Naval Research Laboratory

Washington, DC 20375-5320

NRL/MR/5540--07-9094

A Unified Specification of Behavior for Requirements and Design

JAMES KIRBY, JR.

Center for High Assurance Computer Systems
Information Technology Division

December 10, 2007

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*
10-12-2007	Memorandum Report	Dec. 2002 – Oct. 2007

4. TITLE AND SUBTITLE

A Unified Specification of Behavior for Requirements and Design

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

James Kirby, Jr.

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER
8089

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Research Laboratory, Code 5540
4555 Overlook Avenue, SW
Washington, DC 20375-5320

8. PERFORMING ORGANIZATION REPORT NUMBER

NRL/MR/5540--07-9094

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR / MONITOR'S ACRONYM(S)

11. SPONSOR / MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

It would be useful to write one description of software behavior to serve both requirements and design. Having one description could reduce effort by eliminating the work of developing two descriptions and of keeping them consistent throughout development and maintenance. It would also eliminate the inconsistency inherent in having two descriptions, a fertile source of error. A question paramount to software engineers is Could one description of behavior for a real system serve both requirements and design? The purpose of the present document is to answer that question by producing one such description of the software behavior of a real system. The specification presented here is based upon behavioral specifications extracted from function and abstract interface specifications developed by Paul Clements, Alan Parker, Kathryn Heninger Britton, David Parnas, John Shore, Stuart Faulk, Bruce Labaw, and David Weiss.

15. SUBJECT TERMS

Software behavior
Requirements documentation
Design documentation
Behavioral specification

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
					James Kirby, Jr.
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**	UL	171	**19b. TELEPHONE NUMBER** *(include area code)*
Unclassified	Unclassified	Unclassified			(202) 767-3107

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39.18

Table of Contents

Preface

I Introduction

The typical software development project produces several descriptions of software behavior. Often, natural language provides one description of behavior. UML, data flow diagrams, and pseudocode may provide other descriptions of behavior. Programmers encode software behavior in one or more of a variety of programming languages. Such redundant recording of software behavior is a significant source both of unnecessary work and of error. Descriptions in different languages are difficult to compare with one another for consistency. As discoveries of what the behavior should be are made during development, it is difficult to keep the various descriptions consistent. Often, they are not kept consistent.

In the late 1970s a team at NRL embarked on a project whose goal was to redesign and reimplement the operational flight program for the Navy's A-7E jet aircraft [19]. As part of that project, the team produced an extensive set of development documents, many of which are publicly available. While the A-7E requirements [1] and design documents [4][5][13] are not written in widely differing languages, they provide redundant descriptions of the behavior of the A-7E Operational Flight Program (OFP). This can be seen clearly by comparing the function descriptions in the software requirements document [1] to the function descriptions that specify the function driver module [4]. Both describe the behavior of the A-7E OFP. The former specifies that behavior as the values to be sent to each of the physical output devices as functions of aircraft operating conditions and of inputs received from physical input devices. The latter specifies that behavior as virtual output device programs to be called and values of the programs' input parameters as functions of aircraft operating conditions and of inputs received from virtual input devices.

A previous paper [10] argued that it would be useful to write one description of required software behavior to serve both requirements and design. Having one description could reduce effort by eliminating the work of developing two descriptions and of keeping them consistent throughout development and maintenance. It could also eliminate the inconsistency inherent in having the two descriptions, a fertile source of error.

In reference [10], software behavior is the changes over time of environmental quantities and qualities that the system controls (e.g., when to release a weapon, whether to light an indicator). Mathematical variables denote these quantities and qualities. Mathematical functions, whose domains comprise variables denoting environmental quantities and qualities and variables representing system state, specify the values of those variables. Each of these functions, which can be understood to specify behavior for requirements, also serves design by specifying the behavior of the information hiding module [14] responsible for implementing that function. These modules depend upon other modules with distinct responsibilities. Some are responsible for providing the values of variables denoting environmental quantities and qualities. These modules depend upon modules implementing virtual devices, which interface to the system's physical devices. Some modules provide the values of variables representing system state. Other mod-

ules are responsible for manifesting changes to the system enviroment specified by the functions. These modules also depend upon modules implementing virtual devices.

The definitions of the variables denoting environmental quantities and qualities describe how the values allowed by variable type relate to the particular quantity or quality that variable denotes. These definitions serve requirements by guiding interpretation of the variables and the functions. Annotating with the definitions the programs setting or getting the values of the variables serves design by describing program behavior.

While [10] described a small example applying the ideas, it did not address a question that is paramount to software engineers, Can the ideas be applied to a real system? Answering that question is the purpose of the present report, which includes, beginning on page 1, a *unified* specification of the behavior of the A-7E OFP that is intended to serve both requirements and design. The author adapted the unified specification from design documents for the A-7E OFP [4][5][13]. The unified specification interprets terms defined to describe the behavior of programs implementing virtual devices as variables denoting quantities and qualities in the system environment that the OFP monitors and controls. The author adapted tabular functions describing when to call programs implementing virtual output devices and to what values to set their parameters. The adapted functions specify the values of variables denoting quantities and qualities in the system environment that the OFP controls. The unified specification considers the values that these latter variables assume over time to be the behavior of the OFP.

II Background

The approach taken by the unified specification of behavior of the A-7E OFP is related to that of Heninger [8] (and applied in [1]) and to the Four-Variable Model of Parnas and Madey [17], adopting ideas and terminology from the latter and mechanisms from the former. In the unified specification, the values that a set of variables takes over time describes software behavior. Called *controlled variables*, they denote aspects of the environment that the software controls or affects. A mathematical function, usually tabular in form, gives the value of each of the variables at any point in time. In the domain of the function are *monitored variables* which denote aspects of the environment that the software monitors or measures, *terms* which simplify the specification by representing repeated or complex expressions, and *modes* which abstract system state. While a function may specify the value of more than one variable, the value of each variable is given by exactly one function. In some instances, the function may be broken into distinct pieces that the specification presents together.

In the Four-Variable Model described in [17], Parnas and Madey abstract from the A-7 software requirements model [8]. Instead of using tabular functions to specify required behavior, [17] leaves open the form that the functions describing required behavior of particular systems should take. Mathematical relations on vectors of time functions for monitored, controlled, input, and output variables replace the conditions, events, modes, and tables of the A-7 requirements [8]. The relation

$$\text{REQ: } M \rightarrow C,$$

a relation from all possible histories (where *possible* means allowed by environmental constraints) of the monitored variables to all possible histories of the controlled variables, describes required system behavior. *M*, the domain of *REQ*, is a set of vectors. For each monitored variable, a vector has one element, a time function. The time function, which specifies the value of the monitored variable as a function of time, describes a possible history of that monitored variable. Each vector of monitored variable time functions describes a possible history of all of the monitored variables. *M* is the set of all possible histories of the monitored variables. *C*, the range of *REQ*, is a similar set of vectors of time functions specifying possible histories of the controlled variables. For each possible history of the monitored variables in the set *M*, *REQ* specifies one or more possible histories of the controlled variables in the set *C*. Below, this paper will use similar relations on vectors of time functions for other variables to describe other models.

Similar to *M* and *C*, *I* and *O* are sets of possible histories of the system's input and output devices, respectively. An element from a vector in *I* is a time function representing a possible history of an input received from a particular input device. Similarly, an element from a vector in the *O* is a time function representing a possible history of an output sent to a particular output device. The relation

$$\text{IN: } M \rightarrow I$$

specifies the behavior of the input devices. The relation

$$\text{OUT: } O \rightarrow C$$

specifies the behavior of the output devices.

The A-7E requirements model of [1] and [8], can be represented approximately by the relation

$$REQ_{A\text{-}7E}\text{: } M \times I \times Z \rightarrow O,$$

where *M* represents aircraft operating conditions of the OFP informally described by the *!terms!* of [1], *I* represents physical inputs, and *Z* represents the modes. The relation,

$$Z_f\text{: } M \times I \times Z \rightarrow Z,$$

represents the mode tables of [1].

Reference [10], which motivates the current report, assumes that requirements and design specifications share the relation describing behavior from the Four-Variable Model,

$$\text{REQ: } M \rightarrow C.$$

In addition, the design includes the Device Interface module [2][3] whose interface comprises virtual inputs (I_v)and virtual outputs (O_v). The specification of the Device Interface module records the relations

$$IN_v: M \rightarrow I_v,$$

which describes the behavior of virtual input devices, and

$$OUT_v: O_v \rightarrow C,$$

which describes the behavior of the virtual output devices.

III Reading and Interpreting the Unified Specification of Behavior

Section 1.0 on page 1 through Section 15.0 on page 95 contain the functions, mostly tabular, that specify the values of the controlled variables. For example, Table 3 on page 3 provides the function for the controlled variable !+Aud signal+! which denotes whether the aircraft's audible signal is on steady, off, or beeping. For historical reasons, a bracketing notation indicates variables names and distinguishes different sorts of variables. Table I lists bracketing notation used in this report. The bracketing notation used in !+Aud signal+! indicates that it is a monitored variable, controlled variable, or term defined in the variable dictionary (starting on page 129). In the dictionary entry for !+Aud signal+!, the *C* in the first column indicates that it is a controlled variable. Similarly, *M* or *T* in the first column of an entry distinguish monitored variable and term, respectively. *MC* in the first column of a dictionary entry indicates that the variable is both a monitored and a controlled variable. A variable name bracketed by single exclamation marks (e.g., !A/C facing target!) may be either a term or a monitored variable that is defined in the variable dictionary. The definition of a monitored or controlled variable in the variable dictionary includes the variable's type and its *interpretation*, which describes how the value of the variable relates to the aspect of the environment that the variable denotes. The definition of a term also includes its type. The interpretation of a term may contain either the expression that the term represents or an informal description of its value. A variable name bracketed by double exclamation marks (e.g., !!time beeped!!) is local to a particular function. The function's local dictionary defines the term (see the definition of !!time beeped!! in the local dictionary on page 3) by providing either the expression that the term represents or an informal description of its value. In some instances the interpretation of a term or a local term references a table that specifies its value (see Table 18 on page 15 and Table 154 on page 150). Names bracketed by dollar signs (e.g., On, Off, $Intermittent$) represent the values of enumerated variables. The type dictionary beginning on page 125 defines enumerated and other types. Note that the dollar sign brackets are not consistently used for all enumerated types (e.g., boolean, format, weap_class).

There are several kinds of functions specifying the values of controlled variables. An *event function* specifies the controlled variable !+Aud signal+! (see Table 3 on page 3). It specifies the value the variable assumes when it changes and identifies the events that trigger those changes. The OFP being in one of the modes listed in the *Modes* column of Table 3 selects the row of events that determine when the variable changes value. While the system is in, say, *A/G Guns* mode (modes are discussed below), if the event in the first *Events* column (@T(!+RE pressed+!) WHEN(!+WeaponClass+! = RK)) occurs, then the audible signal goes on steady (indicated by the enumerated value On at the bottom of

TABLE I Bracketing Notation

Brackets	Examples	Interpretation
!+Term+!	!+ Aud signal+!,!+ADI az+!	Monitored variable, controlled variable, or term
!Term!	!A/C facing target!	Monitored variable or term
!!Local term!!	!!time beeped!!	Local term
$Value$	On, Off, $Intermittent$	Value of enumerated variable
Mode, **Mode**	*A/G Guns*, **NBShrike**	Mode
/Input/	/ACAIRB/	Input from physical device
+Program+	+S_AUDIBLE_SIGNAL+	Callable program

the column). The event occurs when the monitored variable !+RE pressed+! goes from false to true (indicated by @T(!+RE pressed+!)) while the monitored variable !+Weapon Class+! has the value RK. In prose, the audible signal goes on steady when the pilot presses the release enable button while the class of the weapon loaded on the currently active weapon station(s) is *rockets*. On the occurrence of either of the two events in the disjunction that the second *Events* column contains, the audible signal goes silent. The first event occurs when the value of the monitored variable !+Rel in Progress+! transitions from false to true. The second event occurs when the monitored variable !+RE pressed+! goes from true to false (i.e., the pilot stops pressing the release enable button). The X in the third event column indicates that audible signal never starts beeping while the system is in the *A/G Guns* mode.

A *condition function* specifies the controlled variable !+ADI az+! (see Table 6 on page 7). The system being in, say, *HUDdown1* mode selects the second row of conditions. Whichever of the three conditions is true—and the rows of conditions are written so that exactly one of the conditions is true—selects which expression at the bottom row of the table determines the value of !+ADI az+!. If the monitored variable !+desig+! is true, then !+ADI az+! assumes the value of the monitored variable !+steering error to tgt+!. Otherwise, whether or not !+Fly to num+! has the value zero determines whether !+ADI az+! assumes the value zero or the value of the local variable !!steering error to ftpt!!. Some condition functions do not always affect software behavior. For such a function (see, e.g., Section 6.4 on page 17), *initiation events* specify when the function begins affecting software behavior and *termination events* specify when the function stops affecting software behavior.

Variable names bracketed by single or double asterisks (e.g., *A/G Guns*, **NBShrike**) represent modes, a grouping of the possible states of the OFP that correspond to aircraft operating conditions [1]. Section 16.0 describes five classes of modes. The system is always in exactly one mode of the Alignment, Test, and Navigation mode classes. Section 16.1 on page 96 specifies which mode that is. The first column of Table 122 on page 96 lists conditions that specify which of the modes the OFP is in on initialization. If the condition in the last row of the first column, NOT !+IMS up+!, is true on initialization, then the OFP is initially in mode *IMS fail*. The remaining tables of Section 16.1 provide rules specifying transitions among modes in the three mode classes. Table 123 on page 97 provides rules for transitioning from modes *Lautocal* and *Sautocal*, listed in

the right most column, to the modes listed in the left most column. Each row of the table specifies a transition rule. The @T or @F in a particular column of a row indicates an event occurring (assuming any guard conditions are satisfied) when the header condition in that column becomes true or false, respectively. A t or f in a column of the row indicates that the header condition in that column is a guard on the event which must be true or false, respectively, for the event to occur. Multiple ts and fs in a row represent a conjunction of conditions. The next to last row of Table 123 specifies that when the OFP is in *Sautocal* on the occurrence of the event @T(!+ND2 stage complete+!) WHEN (NOT (!+Self-test+! = $true) AND !+latitude+! > 80), the OFP transitions to mode *Polarl*.

Section 16.2 on page 111 specifies the initial mode and the transition rules of the Navigation Update mode class. Section 16.3 on page 114 specifies the initial mode and the transition rules of the Weapon Delivery mode class.

IV Creating the Unified Specification of Behavior

Once it is clear how the A-7 design documents capture the behavior of the OFP, extracting that description of behavior to create the unified specification is a conceptually simple exercise. “The Function Driver Module consists of a set of modules called Function Drivers; each Function Driver is the sole controller of a set of closely related outputs. The outputs are either part of a virtual device or provided by the Extended Computer for test purposes. The primary secrets of the Function Driver Module are the rules determining the values of these outputs.”[3] These outputs are the behavior of the OFP. Functions, mostly tabular in form, that capture the rules determining their values specify that behavior.

The functions in the unified specification of behavior and those in the Function Driver Module are alike in that both use the bracketed variable names to specify the events and conditions in the rules determining the values of the outputs. The two sets of functions differ in that functions in the unified specification specify the values of terms identified as controlled variables, while functions in the Function Driver Module specify programs to be called, the values of any parameters to be passed to those programs, and the module (e.g., Device Interface, Shared Services) that specifies each program and parameter. Usually, the module that specifies a program or parameter defines a *!+term+!* describing the output value. The variable dictionary of the unified specification collects such term definitions, labeling them *controlled variables* (or *monitored and controlled variables*, as appropriate). When the module does not provide suitable terms, the author of the unified specification defined them and included them in the variable dictionary, labeling them *controlled variables* (or *monitored and controlled variables*, as appropriate). Table II summarizes the number of controlled variables specified in each section of the unified specification and how many of the definitions came from the A-7E design and how many the author defined. Shaded cells in function tables and the variable dictionary mark such variables and their definitions.

The Function Driver Module provides an index that identifies the module defining each of the bracketed *!+terms+!* that the function drivers’ functions reference (i.e., the *!+terms+!* that are not outputs, see !+RE pressed+! and !+desig+! in the first row of Table 3). The unified specification interprets the bracketed *!+terms+!* as monitored variables if they denote aspects of the environment of the OFP and as terms if they sim-

plify the specification by representing repeated or complex expressions. In either case, the unified specification collects them and their definitions from the defining module into the variable dictionary. In a small number of cases, the function drivers' functions reference *!terms!* defined in the requirements [1]. The unified specification interprets them as monitored variables or terms and collects them in the variable dictionary. In the few instances that module documentation (or the requirements dictionary, in the case of *!terms!*) did not provide suitable definitions of terms, the author of the unified specification defined them and included them in the variable dictionary, labeling them *monitored variables* or *terms*, as appropriate. Shaded cells in the variable dictionary mark such variables and their definitions.

TABLE II Sources of Controlled Variable Definitions

Sections of Functions in Unified Specification	Controlled Variables	Defined in Design	Defined by Author
1.0 Air Data Computer Functions	2	1	1
2.0 Audible Signal Functions	1	1	0
3.0 Computer Fail Signal Functions	1	1	0
4.0 Doppler Radar Set Functions	1	0	1
5.0 Flight Information Display Functions	7	7	0
6.0 Forward Looking Radar Functions	6	6	0
7.0 Head-Up Display Location-Indicator Functions	24	24	0
8.0 Head-Up Display Value-Indicator Functions	7	7	0
9.0 Inertial Measurement Set Functions	27	17	10
10.0 Panel Functions	8	3	5
11.0 Projected Map Display Set Functions	15	8	7
12.0 Shipboard Inertial Navigations System Functions	1	1	0
13.0 Visual Indicator Functions	2	1	0
14.0 Weapon Release System Functions	2	0	3
15.0 Ground Test Functions	1	1	0
Totals	106	79	27

!!Terms!! in the local dictionary of each function driver were, for the most part, copied into the local dictionary of the corresponding function in the unified specification. In some instances, the definition of a *!!term!!* included a call to a program performing some mathematical calculation. In the unified specification, the mathematical calculation replaced the program call in the definition. For example, the local dictionary of the function *Set HUD flight director azimuth position* in [4] defines !!ltd brg ac ftpt!! as the results returned by the program call +SU.LIMIT_2+(!!steering error to ftpt!!, 0.5). (Plus signs bracket the names of programs.) The corresponding function in the unified specification specifies the value of the controlled variable !+FLTDIR azimuth+!. Its local dictionary defines !!ltd brg ac ftpt!! as (!!steering error to ftpt!! / ABS(!steering error to ftpt!!)) x MIN(!!steering error to ftpt!!, 0.5), which is the calculation that the called program performs.

Since the Mode Determination Module hides how to determine the current modes of the OFP [3], the rules specifying initial modes and mode transitions rules don't appear in the design specifications. While the A-7E OFP requirements [1] describe the rules in the format of Section 16.0, the conditions used to construct transition events reference terms defined in the requirements dictionary and inputs from physical devices (indicated by backslash brackets, e.g., /ACAIRB/), rather than the definitions of the variable dictionary in Section 19.0. Consequently, creating the specification of modes of operations in Section 16.0 requires a translation of the conditions used in [1] to conditions based on the variable dictionary in Section 19.0. Table III provides that translation. The first column of the table lists the conditions used in the requirements to describe the mode transition rules. The third column lists the equivalent design condition. which were defined by the author (the fourth column provides explanations). The tables in Section 16.0 which describe initial modes and transition rules are the tables from the requirements with equivalent design expressions from Table III replacing the requirements table headings.

Shaded cells in Table III and in the tables of the unified specifiation (see, e.g., Table 1, Table 24, Table 123, Table 153) indicate terms or expressions that were not present in the A-7E OFP design or that the author changed significantly.

TABLE III A-7 Mode Table Headers and Equivalent Design Expressions ((Sheet 1 of 4))

Terms in A-7 Mode Table Headers	Notes on Requirements	Design Equivalent	Design Definitions (if new !+term+!)	Notes on Design
!/FLYTOTW/ reset!		!Fly to num reset+!	@C(!+Fly to num+!) WHEN (!+Fly to num+! != 0 & PREV(!+Fly to num+!) != 0)	
!/UPDATTW/ = Other!		!+UpdATTW=Other+!	!+Update+! != $Flyover$ & !+Update+! != HUD & !+Update+! != $Radar$ & !+Update+! != $TacL-L$	
!Air velocity test passed!		!+air velocity test passed+!		
!Any dest. entered!	Note abbreviation for "destination"	!+new dest coords entered+!		
!CA stage complete!		!+CA2 stage complete+!		
!CL stage complete!		!+CL2 stage complete+!		
!Data 23! =!Land!		!+land based panel+! = $true$		
!Data 23! = Sea, !Data 23! = !Sea!	Inconsistent usage.	!+land based panel+! = $false$		
!Desig!		!+desig+!		
!Doppler coupled!		!+Doppler coupled+!		
!Doppler up!		!+Doppler up+!		
!FLY-TO changed!		!+Fly to num changed+! OR !+Fly to state changed+!		

TABLE III A-7 Mode Table Headers and Equivalent Design Expressions ((Sheet 2 of 4))

Terms in A-7 Mode Table Headers	Notes on Requirements	Design Equivalent	Design Definitions (if new !+term+!)	Notes on Design
!ground tests finished!		!+ground tests finished+!	@F(!+in Grtest+!)	using generic term !+in x+!
!Guns!		!+Guns+!	!+Weapon Class+! = GN	
!high drag!		!+high drag release+!		
!HS stage complete!		!+HS state complete+!		
!IMS Up!		!+IMS up+!		
!In OFF_MFSW!		!+in Off_MFSW+!		using generic term !+in x+!
!In WD_MFSW!		!+in WD_MFSW+!		using generic term !+in x+!
!Land velocity test passed!		!+land velocity test passed+!		
!latitude! gt 70 deg.		!+latitude+! > 70		
!latitude! gt 80 deg.		!+latitude+! > 80		
!low drag!		!+low drag release+!		
!ND stage complete!		!+ND2 stage complete+!		
!No intervening take-off!		!+No intervening takeoff+!	NOT(@T(!+in flight+!) WHEN (NOT !+in Landaln+!))	My guess.
!Non-zero digit entered!		!+keybd input +! != 0 & !+keybd input+! != $None$		
!Other weapon!		!+Other Weapon+!	!+Station selected+! & !+Weapon Class+! != UN & !+Weapon Class+! != GN & NOT (!+Reserved Weapon+! Or !+Shrike+!)	
!Overfln gt 42nmi!		!+gr ac stik exit+! > 42 nmi.		
!Overflown exit!		!+Overflown exit+!	True if last weapon mode was exited because the target was overflown without a release.	
!Pitch small! AND !Roll small!		!+pitch small+! AND !+roll small+!	!+pitch small+! = ABS(!+pitch system+!) <= 20; !+roll small+! = ABS(!+roll system+!) <= 30	Define new !+terms+!.
!present position entered!		!+new posn entered+!		

TABLE III A-7 Mode Table Headers and Equivalent Design Expressions ((Sheet 3 of 4))

Terms in A-7 Mode Table Headers	Notes on Requirements	Design Equivalent	Design Definitions (if new !+term+!)	Notes on Design
!Ready station!		!+ Ready Station REQ+!	!+Station selected+! & !+Weapon Class+! != UN & !+Weapon Class+! != GN	In the design "ready station" means station selected; in the requirements "ready station" means station selected and it has a weapon.
!Redesignate!		!+Redesignate+!	(!+TD pressed+! OR (!+keybd input+! != 0 & !+keybd input+! != $None$)) & !+in mode AflyUpd+!	Not convinced this is correct.
!Reserved weapon!		!+Reserved Weapon+!	!+Walleye+! OR !+Special+! OR !+Rockets+! OR !+Guns+!	
!Rockets!		!+Rockets+!	!+Weapon Class+! = RK	
!Shrike!		!+Shrike+!	!+Weapon Class+! = SK	
!SINS up!	Not defined in Req.	!+SINS enabled+!		This is my guess.
!SINS velocity test passed!		!+SINS velocity test passed+!		
!Special!		!+Special+!	!!+Weapon class+! = SOD OR !+Weapon class+! = SSH	
!Station selected!		!+Station selected+!	!+nbr rdy sta+! > 0	In the design "ready station" means station selected; in the requirements "ready station" means station selected and it has a weapon.
!Walleye!		!+Walleye+!	!+Weapon Class+! = WL	
!WD MFS!		!+WDMFS+!	!+natt+! OR !+boc+! OR !+ccip+!	!+offset+! is redundant. Left it out.
!Weapon mode! = *BOC*	!Weapon mode! not defined in Req.	!+in BOC+!		using generic term !+in x+!
/ACAIRB/ = Yes		!+in flight+!		
/ENTERSW/ = On		!+Enter pressed+!		
/FLYTOTOG/ = $Dest$		!+Fly to State+! = $Dest$		

TABLE III A-7 Mode Table Headers and Equivalent Design Expressions ((Sheet 4 of 4))

Terms in A-7 Mode Table Headers	Notes on Requirements	Design Equivalent	Design Definitions (if new !+term+!)	Notes on Design
/FLYTOTW/ = 0		!+Fly to num+! = 0		
/GUNSSEL/ = No		NOT !+Gun Enable+!		
/GUNSSEL/ = Yes		!+Gun Enable+!		
/HUDREL/ = Yes		!+HUD reliable+!		
/IMSAUTOC/ = On		!+Auto-cal sw +! = $true$		
/IMSMODE/ = $Gndal$		!+IMS mode+! =$Gndal$		
/IMSMODE/ = $Grid$		!+IMS mode+! =$Grid$		
/IMSMODE/ = $Iner$		!+IMS mode+! =$Iner$		
/IMSMODE/ = $Mag sl$		!+IMS mode+! =$Mag sl$		
/IMSMODE/ = $Norm$		!+IMS mode+! =$Norm$		
/MFSW/ = BOC		!+boc+!		
/MFSW/ = $BOCOFF$		!+boc+! & !+offset+!		
/MFSW/ = $CCIP$		!+ccip+!		
/MFSW/ = $NATT$		!+natt+!		
/MFSW/ = $NATTOFF$		!+natt+! & !+offset+!		
/MFSW/ = $None$		NOT !+boc+! & NOT !+ccip+! & NOT !+natt+! & NOT !+tf+! & NOT !+off-set+!		
/MFSW/=TF		!+tf+!		
/MODEROT/ = $PRE-SPOS$		!+panel mode+! = $Pres-pos$		
/PNLTEST/ = $TEST$		!+Self-test+! = $true$		
/PRESPOS/ = $UPDATE$		!+pres pos+! = $Update$		
/UPDATTW/ = $FLY-OVER$		!+Update+! = $Flyover$		
/UPDATTW/ = HUD		!+Update+! = HUD		
/UPDATTW/ = $RADAR$		!+Update+! = $Radar$		
/UPDATTW/ = $TAC L-L$		!+Update+! = $TacL-L$		

V Discussion

V A A Specification of Required Behavior

In contrast to the Four-Variable Model, the model of [10], and the A-7E requirements discussed above, in the unified specification, the relation

$$REQ_U: I_v \times I \times Z \rightarrow O_v$$

specifies the behavior of the OFP, where I_v (the virtual input variables) and O_v (the virtual output variables) are interpreted as monitored and controlled variables, respectively, I represents physical inputs, and Z represents the modes. The relation,

$$Z_f: I_v \times Z \rightarrow Z,$$

represents the mode tables of Section 16.0 on page 96.

Inspection of the variable dictionary in Section 19.0 finds many entries that clearly denote quantities and qualities in the environment of the OFP, suggesting that it is not unreasonable to interpret them as monitored and controlled variables. For example, the interpretation of the controlled variable !+Aud signal+! is The current state of the audible signal, the interpretation of the monitored variable !+az miss dist+! is The distance along the ground between the target and the ground-projected line from the aircraft to the computed impact point. The interpretations of some variables assume the reader is familiar with concepts and terms described in [1]. For example, understanding the interpretation of !+boresight azimuth+! requires the reader know what the *Ya axis* and the *Xa-Ya plane* are. Instead of just describing how a monitored or controlled variable's value relates to some aspect of the OFP environment, some interpretations also describe how to use the variable, an unwelcome redundancy with the functions that reference the variable. !+E coarse scale+! provides an example of such an interpretation: Scale factor per pulse used for velocity calculation for the Xp axis when the velocities are being measured by the coarse scale.

Functions specify the values that entries that the dictionary identifies as controlled variables must assume as the environment of the OFP changes over time. Definitions of entries that the dictionary identifies as monitored variables and terms and the mode initialization descriptions and mode transition tables describe that environment. This suggests that it is also not unreasonable to interpret the unified specification as a specification of required behavior of the OFP.

The unified specification of the behavior of the A-7E OFP—like the specifications of behavior in the requirements [1] and the design [4]—is *semi-formal*. While much of the notation comprising the specification is formal, there is no formal model underlying the specification. Though formal models for such specifications exist, it would require some work to make this specification adhere to one of them. In addition, many aspects of the specification are informally captured. For example, see the definitions in the local dictionary, Section 6.2.3.

V B A Specification of Behavior for Design

Since the unified specification of behavior was pulled from design specifications of the A-7E OFP with relatively minor editing, it's reasonable to think it can serve design needs. The unified specification of behavior can be incorporated without modification (with some exceptions discussed below) into a design that adheres to the model described in [18] and exemplified by [3]. Such a design consists of a number of information hiding modules [14], some of which provide programs intended to be used by the programs of other modules (see [5] and [13]) and some of which comprise programs, called *function drivers*, that use programs in other modules (see [4]). The function drivers, which specify the values of the controlled variables, use other programs to set the values of the controlled variables and to obtain the values of the monitored variables, terms, and modes that determine what the values of the controlled variables should be. The function drivers are incorporated into the Function Driver Module, a submodule of the Behavior-Hiding Module. Because the organization of the controlled variable functions (function drivers) into Section 1.0 through Section 15.0 reflect an information hiding decomposition of the Function Driver Module, each of the fifteen sections of functions represents a submodule of the Function Driver Module.

Section 16.0 contains the rules specifying the transitions among the system modes, which are secrets of the Mode Determination Module, a submodule of Shared Services Module, which is a sibling of the Function Driver Module. Consequently, Section 16.0 can be thought of as part of the module's internal design, specifying how to implement the module's functions, as opposed to specifying their black box behavior.

The definitions in the variable dictionary of Section 19.0 can be used to describe the behavior of programs in the submodules of the Device Interface Module that implement virtual devices, and of programs on the interfaces of various submodules of the Shared Service Module. Associating a controlled variable with the input parameter to a program indicates that the effect of calling the program with the parameter set to a particular value is to affect the environmental aspect denoted by the controlled variable in the appropriate way. For example, +S_AUDIBLE_SIGNAL+ (plus signs bracket the names of programs) is a program on the interface of the Audible Signal device interface module (Table IV illustrates documentation of the program adapted from [13]). It has one input parameter of type AUD_ind_cntrl (which can have values On, Off, and $Intermittent$). The controlled variable !+Aud signal+!, which denotes the "current state of the audible signal", describes the effect of setting the input parameter. Thus, the effect of calling the program +S_AUDIBLE_SIGNAL+ with input parameter p1 is to set the value of the controlled variable !+Aud signal+! to the value of p1, causing the audible signal either to be silent, to be on steady, or to beep. It is the responsibility of the implementation of the function driver that specifies the controlled variable !+Aud signal+! (Table 3 on page 3) to call +S_AUDIBLE_SIGNAL+ and pass it the parameter value specified by the function.

TABLE IV Audible Signal Module Access Program Table

Program	Parameters	Description
+S_AUDIBLE_SIGNAL+	p1: AUD_ind_cntrl; I	!+Aud Signal+!

Similarly, associating a monitored variable with the output parameter of a program indicates that on return from a call to the program, the output parameter's value will reflect appropriately the environmental aspect denoted by the monitored variable. For example, +G_WEAPON_RELEASE_CLASS+ is a program on the interface of the Weapon Characteristic Submodule of the Device Interface Module (Table V illustrates documentation of the program adapted from [13]). It has one output parameter of type weap_class. The monitored variable !+Weapon Class+! which is the "class of the weapon loaded on the currently active weapon station(s)" describes the value returned by the parameter. On return from a call to +G_WEAPON_RELEASE_CLASS+, the parameter has the value, say, RK if and only if rockets are loaded on the currently active weapon station(s). The implementation of the function driver that specifies the controlled variable !+Aud signal+! (see the fourth row of Table 3 on page 3) can call the program to determine whether to set the audible signal on steady when it detects that the pilot has pressed the release enable button. The function detects the latter by relying on the Weapon Release Submodule of the Device Interface Module to signal occurrence of the event @T(!+RE pressed+!).

TABLE V Weapon Characteristics Module Access Program Table

Program	Parameters	Description
+G_WEAPON_RELEASE_CLASS+	p1: weap_class; O	!+Weapon Class+!

Each function driver implementation, then, uses the appropriate program to set each controlled variable whose value it determines. Similarly, the function driver implementation uses the appropriate program to obtain the value of each monitored variable, term, and mode that it references. Other modules (see [3]) provide the programs that set the values of controlled variables and obtain the values of monitored variables, terms, and modes. In the case of events, these modules provide special programs that signal when the value of a variable of interest changes.

The remainder of this section discusses functions that presented a challenge to the model, in particular, those specifying values of controlled variables defined by the author (see column 4 of Table II). In some instances, the Device Interface module that provides the facilities for setting the controlled variable poses the challenge. The discussion is organized by function driver module.

Air Data Computer. The function in Table 1 on page 1 describes rules for setting the value of controlled variable !+sea level pressure+!, which the variable dictionary (Section 19.0 on page 129) defines as Atmospheric pressure at sea level. It is unintuitive, at least, to think of the aircraft controlling atmospheric pressure as Table 1 specifies. Since this is a value that the Air Data Computer (ADC) virtual device requires, it would be more sensible to treat !+sea level pressure+! as a monitored variable that the ADC uses the function driver to obtain.

Doppler Radar Set. The Doppler Radar Set (DRS) requires a minor wrinkle to the model we've been following. Because the DRS module provides two parameterless programs to turn the DRS on (+START_DRS+) and off (+STOP_DRS+), there was no need for the module to define a term describing parameters. Rather than describing the value the

function driver should pass to a DRS program, the value that Table 5 assigns to !+DRS on+! determines which program to call.The DRS Device Interface module could document the programs as Table VI illustrates. The reader should interpret the last column of each for row of the table to mean the value to which a call to the program in the first column of the row sets the controlled variable !+DRS on+!.

TABLE VI Doppler Radar Set Module Access Program Table

Program	Parameters	Description
+START_DRS+		!+DRS on+! = TRUE
+STOP_DRS+		!+DRS on+! = FALSE

Head-Up Display Location-Indicator. For some HUD monitored variables, the Device Interface module provides one program that will set several of them with one call. There are a number of *symbols* that the HUD can display to the pilot (e.g., aiming symbol, in-range cue, flight path maker), each of which can be in one of a variety modes (e.g., on, off, blinking). Rather than providing a distinct program for each symbol that will set its mode, the Device Interface module provides +S_HUD_SYMBOL_MODE+ that, given a list of symbol names and a mode, will display all the symbols as specified. This allows the software to take advantage of a hardware capability for controlling certain symbols together [13], but complicates using the unified specification of behavior in the design.

For a HUD symbol whose position requires several dimensions to specify (e.g., elevation, azimuth, rotation), the Device Interface module provides one program to set its position that accepts all the parameters required to specify that position. The function drivers for such symbols specify the positions together, though not always in the same table.

Inertial Measurement Set. Like !+sea level pressure+! and ADC, the ten controlled variables defined by the author whose values are set by the Inertial Measurement Set (IMS) Functions have to do with initializing a device. The definitions come from the description of effects of calling the various IMS programs in the IMS device interface module. It would probably be more sensible to treat the controlled variables as monitored variables that the IMS Device Interface module uses the function drivers to obtain.

Panel. The aircraft provides a one character pilot display, *mark window*, and two general purpose pilot displays, a seven-digit *lower window* and a six-digit *upper window*, which display a large variety of information, e.g., position of the aircraft, altitude, hardware diagnostic information. While the Panel Device Interface module provides a number of programs for setting and clearing the upper and lower windows, the Panel function drivers use analogous programs, which better support formatting of the upper and lower windows, provided by the Shared Services module. The author-defined controlled variables !+up win fmt+!, !+up win val+!, !+low win fmt+!, and !+low win val+! and the tables which assign them values (Table 85 on page 66 and Table 86 on page 67) represent a minor formalization and reorganization of concepts in the corresponding function driver module. Shared Services provides several distinct programs for setting each of the upper and lower windows, respectively. The value of the corresponding window for-

mat (!+up win fmt+! or !+low win fmt+!) determines which of the programs the implementation calls for displaying the window value contained in !+up win val+! or !+low win val+!.

The Panel Device Interface module provides two programs to control the mark window display whose value the (author-defined) controlled variable !+Mark window+! determines. One, the parameterless +CLEAR_MARK+, blanks out the mark window display. The second, +S_MARK_WINDOW+, displays in the mark window the alphanumeric character provided by its one parameter. When the function in Table 82 on page 63 sets the value of !+Mark window+! to blank, the implementation calls +CLEAR_MARK+. When the function in Table 83 sets the value of !+Mark window+!, the implementation calls +S_MARK_WINDOW+, passing it that value.

There are several cases that the author has not fitted into the design model. One of them has to do with the local variable !!North light!! which appears in the right hand column of Table 85. The definition of !!North light!! calls for setting the values of two controlled variables, !+N Light+! and !+Format U321+!, to $true$. The two controlled variables should have one or more distinct functions specifying their values. The local variable !!Compfail!!, which appears in Table 86, is another problematic case. Its definition calls for the setting of several outputs controlling lights on the upper and lower windows, which should also have one or more distinct functions specifying their values. While the author has not resolved them, none of these cases appears to pose a particular challenge to using the unified specification of behavior in the design.

Projected Map Display Set. One of the most challenging function drivers of the A-7E to accommodate to the model of the unified specification was that for positioning the map in Section 11.5 on page 84. The challenge was driven by the programs provided by the Projected Map Display Set (PMDS) Module of Device Interface. Given the latitude and longitude of a point on the earth, the PMDS program +G_MAP_DISPLAYABLE+ returns the boolean !+Map displayable+! indicating whether the map can display that point. The following call to +S_MAP_POSITION+ positions the map to display that point if +G_MAP_DISPLAYABLE+ returned true. If +G_MAP_DISPLAYABLE+ returned false, then +S_MAP_POSITION+ invokes the PMDS program +DISPLAY_MAP_WARNING+ which displays a distinctive warning display on the map screen. The function driver calls +DISPLAY_MAP_WARNING+ when +G_MAP_DISPLAYABLE+ returns false.

To fit this function driver into the unified specification model, the author defined several terms and refined several others. New to the unified specification, the boolean controlled variable !+Map warning+! is true iff a distinctive warning display appears on the map screen, as after a call to +DISPLAY_MAP_WARNING+. It is false after a call to +S_MAP_POSITION+ when !+Map displayable+! is true. Definitions of the new terms !!Recalled lat!! and !!Recalled long!! formalize information already present in the function driver's local dictionary. The unified specification includes refined definitions of !+Map displayable+!, replacing "requested location" by local terms !!refpt lat!! and !!refpt long!!, and of !!Position displayable!!, replacing the call to +G_MAP_DISPLAYABLE+, with !+Map displayable+!, whose value the program returns. While Table VII illustrates how the PMDS Device Interface could document the program +G_MAP_DISPLAYABLE+, it is not clear

how to use the unified specification of behavior to document +S_MAP_POSITION+, whose behavior depends upon the previous call to +G_MAP_DISPLAYABLE+.

TABLE VII Projected Map Display Set Module Access Program Table

Program	Parameters	Description
+DISPLAY_MAP_WARNING+		!+Map warning+! = $true$

The six remaining author-defined controlled variables represent a formalization of function drivers for setting the map reconfiguration values (see Section 11.7 on page 87).

Weapon Release System. The Weapon Release System function drivers are also challenging. One function specifies when to call the parameterless program +PREPARE_WEAPON+. The unified specification defines the boolean controlled variable !+prepare weapon+!. Toggling the value of !+prepare weapon+! signals when to commence preparing weapons on the current weapon station(s) for release, which is what a call to +PREPARE_WEAPON+ accomplishes.

A second function driver determines when to call the program +RELEASE_WEAPON+, which issues the fire ready and bomb release signals for a length of time specified by the program's input parameter. As with +PREPARE_WEAPON+, the unified specification defines a boolean controlled variable !+release weapon+!. Toggling the value of !+release weapon+! indicates when to issue the signals. Contrary to practice in other programs the author has examined in the Device Interface module, !+release pulse width+!, the term annotating the input parameter of +RELEASE_WEAPON+, is defined in another submodule of the Device Interface module, which provides its value for the current weapon station(s). For this reason, the unified specification considers the term !+release pulse width+! to be a monitored variable. The unified specification defines the controlled variable !+set release pulse width+!, whose value a function driver sets (see Table 120 on page 94) and which should annotate the input parameter of +RELEASE_WEAPON+, replacing !+release pulse width+!.

VI Summary

The purpose of the present document is to report on an attempt to produce one description of the software behavior of a real system that could serve both requirements and design as described by [10]. The specification presented here was adapted from *design* documentation. Terms defined to describe the behavior of programs providing virtual devices filled in for monitored and controlled variables denoting quantities and qualities in the system environment. The author adapted tabular functions describing when to call programs implementing virtual output devices and to what values to set their parameters. The adapted functions specified the values of virtual outputs that the unified specification interprets as controlled variables. Of the 106 controlled variables, only a handful was the author unable to fit into the model of [10]. In particular, it was not clear how to use the unified specification of behavior to document the behavior of +S_MAP_POSITION+ in the PMDS module and the behavior of +S_HUD_SYMBOL_MODE+ in the HUD Location-indicator module.

Acknowledgements. The author is grateful to Thomas Alspaugh for comments and suggestions that led to improvements in the report.

VII References

[1] T.A. Alspaugh, S.R. Faulk, K.H. Britton, R.A. Parker, D.L. Parnas, J.E. Shore, *Software Requirements for the A-7E Aircraft*, NRL Final Report 5530, 1992.

[2] K. Britton, R. Parker, D. Parnas. "A Procedure for Designing Abstract Interfaces for Device Interface Modules", *Proc., 5th International Conf. on Software Eng*., March 1981, pp.195-204.

[3] K.H. Britton, D.L. Parnas, *A-7E Software Module Guide*, NRL Memorandum Report 4702, 1981.

[4] P.C. Clements, *Function Specifications for the A-7E Function Driver Module,* NRL Memorandum Report 4658, November 27, 1981.

[5] P.C. Clements, *Abstract Interface Specifications for the A-7E Shared Services Module,* NRL Memorandum Report 4863, 1982.

[6] P.C. Clements, *Abstract Interface Specifications for the Numerical Algorithms Submodule of the A-7E Software Utility Module,* unpublished, undated report.

[7] P.C. Clements, S.R. Faulk, D.L. Parnas, *Interface Specifications for the SCR (A-7E) Application Data Types Module*, NRL Memorandum Report 8734, August 23. 1983.

[8] K.L. Heninger. "Specifying Software Requirements for Complex Systems: New Techniques and their Application", *IEEE Trans. on Software Engineering*, IEEE Computer Society, January 1980, pp. 2-13.

[9] J. Kirby, "Model-Driven Agile Development of Reactive Multi-Agent Systems", *Proc., 30th International Computer Software and Applications Conference (COMPSAC 2006)*, September 2006, Chicago, IL.

[10] J. Kirby, "Rewriting Requirements for Design". *Proceedings, IASTED International Conference on Software Engineering and Applications (SEA) 2002*, November 4-6, 2002, Cambridge, MA, USA.

[11] J. Kirby, J. Archer, C. Heitmeyer. "SCR: A Practical Approach to Building a High Assurance COMSEC System", *Proceedings, Annual Computer Security Applications Conference (ACSAC) '99*, December 1999.

[12] B.G. Labaw, *Abstract Interface Specifications for the Physical Models Module,* NRL Memorandum Report 7595-021, September 9, 1986.

[13] A. Parker, K.H. Britton, D. Parnas, J. Shore, *Abstract Interface Specifications for the A-7E Device Interface Module*, NRL Memorandum Report 4385, November 20, 1980.

[14] D.L., Parnas, "On the Criteria to be Used in Decomposing Systems in Modules", *Comm. of the ACM*, 15(12), 1972, 1053-1058.

[15] D.L. Parnas, K.H. Britton, D.M. Weiss, P.C. Clements, *Interface Specifications for the SCR (A-7E) Extended Computer Module*, NRL Memorandum Report 5502, December 31, 1984.

[16] D.L. Parnas and P.C. Clements, "A Rational Design Process: How and Why to Fake It", *IEEE Trans. on Software Engineering*, IEEE Computer Society, February 1986, pp. 251-257.

[17] D.L. Parnas and J. Madey, "Functional Documentation for Computer Systems Engineering", *Science of Computer Programming*, Elsevier, October 1995, pp 41-61.

[18] D. Parnas, P. Clements, and D. Weiss. "The Modular Structure of Complex Systems", *IEEE Trans. on Software Engineering*, IEEE Computer Society, March 1985.

[19] D.M. Weiss, Introduction to "The Modular Structure of Complex Systems", *Software Fundamentals, Collected Papers of David L. Parnas*, ed. M.D. Hoffman and D.M. Weiss, Addison-Wesley 2001.

Unified Behavioral Specification of the A-7E OFP

James Kirby, Jr.

Behavioral specifications extracted from function and abstract interface specifications developed by Paul Clements, Alan Parker, Kathryn Heninger Britton, David Parnas, John Shore, Stuart Faulk, Bruce Labaw, and David Weiss.

1.0 Air Data Computer Functions

1.1 Set ADC estimate of sea level pressure

TABLE 1. Set ADC estimate of sea level pressure

MODES	EVENTS	
All modes	@T(!+Fly to state+! = $Dest$) WHEN(!+Fly to num+! != 0) OR @T(!+Init complete+!) WHEN(!!destnot0!!) OR @T(!!flyto nonzero!!) WHEN(!+Fly to state+! = $Dest$) OR @T(!+new dest mslp pnl entered+!) WHEN(!+dest entry pnl+! = !+Fly to num+!)	@F(!!destnot0!!) OR @T(!+Init complete+!) WHEN(NOT !!destnot0!!)
!+sea level pressure+!	!!dest mslp!!	29.92” Hg

1.1.1 Local Dictionary

!!dest mslp!! !+dest mslp pnl+!, indexed by !+Fly to num+!.

!!destnot0!! !+Fly to state+! = $Dest$ AND !+Fly to num+! !=0

!!flyto nonzero!! Occurs whenever @T(!+Fly to num changed+!) occurs and the new value != 0; the previous value may or may not have been zero.

Manuscript approved November 21, 2007.

1.2 Set L-probe switch

TABLE 2. Set L-probe switch

MODES	EVENTS
All navigation and alignment modes	@T(!+new L-probe pnl entered+!)
!+L-probe+!	!+L-probe pnl+!

2.0 Audible Signal Functions

2.1 Audible signal mode

TABLE 3. Audible signal mode

MODES	EVENTS		
NBShrike	@T(!+RE pressed+!) WHEN(!+desig+!)	@F(!+RE pressed+!) OR @T(!+time to prepare+!) WHEN(!+desig+!)	X
NBnot- Shrike	@T(!+RE pressed+!) WHEN(!+desig+!)	@T(!+Rel in Progress+!) OR @F(!+RE pressed+!)	X
A/A Manrip *CCIP* *Manrip*	@T(!+RE pressed+!)	@F(!+RE pressed+!)	X
A/G Guns	@T(!+RE pressed+!) WHEN(!+Weapon Class+! = RK)	@T(!+Rel in Progress+!) OR @F(!+RE pressed+!)	X
Walleye	@T(!+RE pressed+!)	@T(!!time tone on!! >= 1 sec)	X
LoNuke	@T(!+rmax+!) OR @T(!+rmin+!) OR @T(!+RE pressed+!) WHEN(!+desig+!)	@T(!!time since rmax!! >= 1 sec) OR @T(!+rmin+6000+!) OR @T(!+pitch IMS+! > 15 deg AND NOT !+RE pressed+!) OR @T(!+Rel in Progress+!) OR @T(!!time beeped!! > 2 sec)	@T(!+r65+!)
HiNuke	@T(!+RE pressed+!) WHEN(!+desig+!)	@F(!+RE pressed+!) OR @T(!+Rel in Progress+!) OR @T(!!time beeped!! > 2 sec)	X
!+Aud signal+!	On	Off	$Intermittent$

2.1.1 Local Dictionary

!!time beeped!! Elapsed time since @T(!+Aud signal+! = $Intermittent$)

!!time since rmax!! Elapsed time since @T(!+rmax+!) occurred.

!!time tone on!! Elapsed time since @T(!+Aud signal+! = On)

2.2 Set the audible signal beep period

This function is never performed, because under the current requirements, the beep period never needs to be changed from its default value.

3.0 Computer Fail Signal Functions

3.1 Computer Fail Signal

TABLE 4. Computer fail signal

MODES	EVENTS	
All modes	@T(!+failed state+!)	@T(!+Init complete+!)
!+Comp fail+!	true	false

4.0 Doppler Radar Set Functions

4.1 Start/stop the Doppler radar

TABLE 5. Start/stop the Doppler radar

MODES	EVENTS	
All modes	@T(!+in flight+!)	@F(!+in flight+!)
!+DRS on+!	TRUE	FALSE

5.0 Flight Information Display Functions

5.1 Set ADI azimuth indicator

TABLE 6. Set ADI azimuth indicator

MODES	CONDITIONS		
No weapon mode listed below	NOT !+in flight+! OR !+Fly to num+! = 0	!+in flight+! AND !+Fly to num+! != 0	X
HUDdown1 *Nattack* *SHUDdown1* *Snattack*	NOT !+desig+! AND !+Fly to num+! = 0	NOT !+desig+! AND !+Fly to num+! != 0	!+desig+!
BOCoffset *HUDdown2* *Noffset* *SBOCoffset* *SHUDdown2* *Snoffset*	X	NOT !+desig+!	!+desig+!
BOC *SBOC*	X	X	Always
BOCFlyto0 *SBOCFlyto0*	NOT !+desig+!	X	!+desig+!
!+ADI az+!	0	!!steering error to ftpt!!	!+steering error to tgt+!

The value of !+ADI az+! must be limited to within +/- 2.5 degrees.

5.1.1 Local Dictionary

!!steering error to ftpt!! !+brg grtk ftpt+! modulo 360 - 360k, where k = 0 if !+brg grtk ftpt+! modulo 360 <= 180 and k = 1 otherwise.

5.2 ADI elevation indicator

5.2.1 Initiation/Termination Events

Initiation. @T(!+ADI elev in view+!)

Termination. @F(!+ADI elev in view+!) OR @F(!+ADI elev avail+!)

TABLE 7. ADI elevation in view

MODES	EVENTS	
IMS fail **NBShrike** All alignment modes except *Air-aln*	X	@T(In mode) WHEN(!+ADI elev avail+!)
NBnotShrike **HiNuke** *Walleye*	@T(ABS(!+LSC elevation+!) <= 4 AND !+ADI elev avail+!)	@T(ABS(!+LSC elevation+!) > 4) WHEN(!+ADI elev avail+!) OR @F(!+ADI elev avail+!)
LoNuke	@T(ABS(!+PUAC elevation+!) <= 4 AND !+ADI elev avail+!)	@T(ABS(!+PUAC elevation+!)> 4) WHEN(!+ADI elev avail+!) OR @F(!+ADI elev avail+!)
!+ADI elev in view+!	TRUE	FALSE

TABLE 8. ADI elevation indicator

MODES	CONDITIONS		
HiNuke **NBnot-Shrike** *Walleye*	Always	X	X
LoNuke	X	NOT !!ac inverted!!	!!ac inverted!!
!+ADI elev+!	!+LSC elevation+!	!+PUAC elevation+!	- !+PUAC elevation+!

5.2.2 Local Dictionary

!!ac inverted!! ABS(!+roll IMS+!) > 90

5.3 Set HSI pointer 1

TABLE 9. Setting the HSI-1 and DME displays except *Grtest*

MODES	CONDITIONS	
All alignment and navigation modes with no modes listed below;	!+Fly to num+! = 0	!+Fly to num+! != 0
HUDdown1 *Nattack* *SHUDdown1* *Snattack*	NOT !+desig+! AND !+Fly to num+! = 0	!+desig+! OR !+Fly to num+! != 0
BOC *BOCoffset* *HUDdown2* *Noffset* *SBOC* *SBOCoffset* *SHUDdown2* *Snoffset*	X	Always
BOCFlyto0 *SBOCFlyto0*	NOT !+desig+!	!+desig+!
!+HSI 1+!	0	!!brg!!
!+DME display+!	0	!!DME integer display!!

TABLE 10. Setting the HSI-1 and DME displays for *Grtest

MODES	CONDITIONS	
Grtest	!+test stage+! = $AC1$	!+test stage+! = $AC2$
!+HSI 1+!	0	225
!+DME display+!	555	553

Note: The DME must be set to 555 actually in !DC Tstage! to allow the servos to settle in time for !AC1 Tstage!.

TABLE 11. When to display/remove the DME flag (and units of display)

MODES	EVENTS		
All navigation or alignment modes when not in a mode listed below	@T(!!range!! >= 1000nmi AND in mode)	@T(!!range! !< 1000nmi AND in mode)	X
BOC *BOCFlyto0* *BOCoffset* *HUDdown1* *HUDdown2* *Nattack* *Noffset* *SBOC* *SBOCFlyto0* *SBOCoffset* *SHUDdown1* *SHUDdown2* *Snattack* *Snoffset*	X	@T(!+desig+! AND !!range!! >= 10nmi AND in mode) OR @F(!+desig+! AND in mode)	@T(!+desig+! AND !!range!!<10 nmi) AND in mode)
!+DME flag+!	TRUE	FALSE	TRUE
!!units!!	nmi	nmi	thousands of ft

5.3.1 Local Dictionary

!!brg!! !+brg grtk x+! where x is replaced by the abbreviation for the current !!refpt!!.

!!DME integer display!! The value of !!range!!, converted to an integer. Whether the integer should be the number of feet or nautical miles in the distance is determined by the definition of !!units!! in the table above.

!!range!! !+gr ac x+! where x is replaced by the abbreviation for the current !!refpt!!.

!!refpt!! Defined by table below. The mnemonic abbreviations are also given in parentheses for each reference point. Thus, for instance, when !!refpt!! is defined to be the target, the definition of !!brg!! is !+brg grtk tgt+!.

TABLE 12. Definition of !!refpt!!

MODES	CONDITIONS		
All alignment and navigation modes with no mode listed in other rows;	Always	X	X
HUDdown1 *Nattack* *SHUDdown1* *Snattack*	NOT !+desig+!	X	!+desig+!
HUDdown2 *Noffset* *SHUDdown2* *Snoffset*	NOT !+desig+! AND NOT !+after slewing+!	NOT !+desig+! AND !+after slewing+!	!+desig+!
BOC *SBOC*	NOT !+after slewing+! OR !+gr ac ftpt+! > 30 nmi	X	!+after slewing+! AND !+gr ac ftpt+! <= 30 nmi
BOCFlyto0 *SBOCFlyto0*	X	X	Always
BOCoffset *SBOCoffset*	!+gr ac ftpt+! > 30 nmi AND NOT !+desig+!	!+gr ac ftpt+! <= 30 nmi AND NOT !+desig+!	!+desig+!
!!refpt!!	!Fly-to-point!	!OAP!	!target!
Abbreviation:	ftpt	oap	tgt

!!units!! Defined by Table 11 on page 10.

5.4 Set HSI Pointer #2

5.4.1 Initiation/Termination Events

TABLE 13. Initiation/Termination of !+HSI2+!

MODES	Initiation events	Termination events
All align and nav modes except *SINSaln*	@T(!+in flight+!)	@F(!+in flight+!)
SINSaln	@T(In mode)	@F(!+align stage+! = CA)

5.4.2 Function Definition

TABLE 14. Value of !+HIS2+!

MODES	!+HSI 2+!
SINSaln	Oscillated between 0 and 11.3 each second; displays 11.3 for .6 seconds, and then 0 for .4 seconds.
All alignment and nav modes except *SINSaln*	!+grtk+!

6.0 Forward Looking Radar Functions

6.1 Set FLR Mode

TABLE 15. Set FLR mode

MODES	EVENTS		
RadarUpd	@T(In mode AND !+gr ac fxpt+! <= 22 nmi AND !+FLR mode+! != TF)	X	X
BOC *BOCoffset* *SBOC* *SBOCoffset*	@T(In mode AND !+gr ac ftpt+! <= 20 nmi AND !+FLR mode+! != TF)	X	X
BOCFlyto0 *SBOCFlyto0*	@T(In mode AND !+FLR mode+! != TF)	X	X
No other listed mode	X	X	@T(In mode AND !+FLR mode+! != TF)
HUDUpd *CCIP* *HUDdown1* *HUDdown2* *Nattack* *Noffset* *SHUDdown1* *SHUDdown2* *Snattack* *Snoffset*	X	@T(In mode AND !+FLR mode+! != TF)	X
A/G Guns	X	@T(!+Gun Enable+! AND !+FLR mode+! != TF) OR @T(In mode AND !+Weapon Class+! = RK AND !+FLR mode+! != TF)	@F(In mode WHEN (!+FLR mode+! != TF)
!+FLR mode+!	$CDCE$	$Ranging$	$Idle$

Before changing mode from $CDCE$ to $Ranging$ or vice versa, the mode must first be set to $Idle$.

6.2 Position the FLR azimuth and range cursors

6.2.1 Initiation/Termination Events

For placing the FLR azimuth cursor:

Initiation. @T(!+FLR az cursor mode+! = On)

Termination. @T(!+FLR az cursor mode+! = Off)

For placing the FLR range cursor:

Initiation. @T(!+FLR mode+! = $CDCE$)

Termination. @F(!+FLR mode+! = $CDCE$)

6.2.2 Function Definition

TABLE 16. Position the FLR azimuth and range cursors

MODES	CONDITIONS		
BOC *BOCoffset* *SBOC* *SBOCoffset* *RadarUpd*	NOT !+during slewing+!	!+during slewing+!	X
BOCflyto0 *SBOCflyto0*	!+desig+! AND NOT !+during slewing+!	!+during slewing+!	NOT !+desig+! AND NOT !+during slewing+!
!+FLR az cursor posn+!	!!ltd brg grtk refpt!!	!!Az slew posn!!	0
!+Rng cursor+!	!+sr ac (!!refpt!!)+!	!!Rng slew posn!!	8 nmi

6.2.3 Local Dictionary

!!Az slew posn!! The new position of the azimuth cursor, computed by adding !+slew FLR delta az+! to the previous position of the azimuth cursor.

!!Rng slew posn!! The new position of the range cursor, computed by adding !+slew FLR delta rng+! to the previous position of the range cursor.

!!ltd brg grtk refpt!! Under some circumstances, the FLR azimuth cursor is positioned at the left or right screen edge, as defined in Table 17.

TABLE 17. The value of !!ltd brg grtk refpt!!

!+brg grtk (!!refpt!!)+! (in degrees)	!!ltd brg grtk refpt!!
>= 270 AND <= 315	!+Az cursor lft max+!
> 45 AND <= 90	!+Az cursor rgt max+!
> 315 OR < 45	!+brg grtk (!!refpt!!)+!

If the !!refpt!! is the imaginary point 8 nmi ahead of the a/c on the ground track, then !+brg grtk (!!refpt!!)+! is considered to be 0.

!!refpt!! Defined by Table 18. The mnemonic abbreviations are also given in parentheses for each reference point. Thus, for instance, when !!refpt!! is defined to be !target!, the quantity !+sr ac (!!refpt!!)+! actually refers to !+sr ac tgt+!.

TABLE 18. The value of !!refpt!!

MODES	CONDITIONS					
BOC *SBOC*	NOT !+desig+!	!+desig+!	X	X	X	X
BOCflyto0 *SBOCflyto0*	X	!+desig+!	X	X	X	NOT !+desig+!
BOCoffset *SBOCoffset*	!+before slewing+! AND NOT !+desig+!	X	!+desig+! OR !+after slewing+!	X	X	X
RadarUpd	X	X	X	!+before slewing+!	!+after slewing+!	X
!!refpt!!:	fly-to- point (ftpt)	target (tgt)	offset aimpoint (oap)	called- up point (cup)	fix point (fxpt)	a point 8 nmi ahead of a/c on ground track

6.3 Set FLR azimuth cursor mode

TABLE 19. Set FLR azimuth cursor mode

MODES	EVENTS		
BOC *BOCoffset* *SBOC* *SBOCoffset* *RadarUpd*	@T(In mode AND NOT !!refpt ahead!! AND !+FLR mode+! !=TF)	@T(!+FLR mode+! = $CDCE$ AND !!refpt ahead!!)	X
BOCflyto0 *SBOCflyto0*	@T(!+desig+! AND NOT !!refpt ahead!! AND !+FLR mode+! !=TF)	@T(!+FLR mode+! = $CDCE$ AND !!refpt ahead!!)	X
!+FLR az cursor mode+!	Off	On	$Intermittent$

6.3.1 Local Dictionary

!!refpt!! Defined in the previous function.

!!refpt ahead!! !+*x* ahead+! where *x* is replaced by the abbreviation of the current !!refpt!!.

6.4 FLR elevation and azimuth

6.4.1 Initiation/Termination Events

Initiation. @T(!+FLR mode+! = $Ranging$)

Termination. @F(!+FLR mode+! = $Ranging$)

TABLE 20. FLR elevation and azimuth

MODES	!+FLR elev+!	!+ FLR az+!
HUDUpd *A/G Guns* *HUDdown1* *HUDdown2* *Nattack* *Noffset* *SHUDdown1* *SHUDdown2* *Snattack* *Snoffset*	!!FLR AS elev!!	!+AS azimuth+!
CCIP	!!FLR LSC elev!!	!+LSC azimuth+!

6.4.2 Local Dictionary

!!FLR AS elev!! The angle whose tangent is the product of the tangent of the !+AS elevation+! and the cos(!+AS azimuth+!).

!!FLR LSC elev!! The angle whose tangent is the product of the tangent of the !+LSC elevation+! and the cos(!+LSC azimuth+!).

6.5 Set the FLR symbol blink period

This function is never performed, because under the current requirements, the FLR symbol blink period need never be changed from its default value.

7.0 Head-Up Display Location-Indicator Functions

7.1 Set HUD aiming symbol mode

TABLE 21. Set HUD aiming symbol mode

MODES	EVENTS		
Landaln *01Update* *I* *OLB* *PolarI*	@T(!+aiming switches+!)	X	X
A/A Guns *A/A Manrip* *A/G Guns* *BOCFlyto0* *HUDdown1* *HUDdown2* *Nattack* *Noffset* *SBOCFlyto0* *SHUDdown1* *SHUDdown2* *Snattack* *Snoffset*	@T(In mode)	X	X
BOC *BOCoffset*	@T(In mode AND !+gr ac HUDrefpt+! <= 30 nmi)	@T(In mode AND !+gr ac HUDrefpt+! > 30 nmi)	X
SBOC *SBOCoffset*	@T(In mode AND !+gr ac HUDrefpt+! <= 30 nmi)	@T(In mode AND !+gr ac HUDrefpt+! > 42 nmi)	X
RadarUpd	@T(In mode AND !+gr ac HUDrefpt+! <= 20 nmi)	@T(In mode AND !+gr ac HUDrefpt+! > 20 nmi)	X
HUDUpd	@T(In mode AND !+gr ac HUDrefpt+! <= 22 nmi)	@T(In mode AND !+gr ac HUDrefpt+! > 22nmi)	X
Walleye	@T(In mode)	@T(!+RE pressed+!)	X
No other mode listed above	X	@T(In mode)	X
!+AS mode+!	On	Off	$Intermittent$

7.2 Set HUD aiming symbol position

TABLE 22. Set HUD aiming symbol position

MODES	CONDITIONS		
HUDaln *Landaln* *01Update* *I*, *OLB*, *PolarI*	!+after slewing+!	!+before slewing+!	!+during slewing+!
!+AS elevation+!	!+HUDrefpt elev+!	0 deg.	!!Slewed AS elev!!
!+AS azimuth+!	!+HUDrefpt az+!	0 deg.	!!Slewed AS az!!

TABLE 23. Where to position the HUD aiming symbol in weapons and update modes

MODES	CONDITIONS			
Nattack	!!SK!! AND NOT !+desig+! AND NOT !!slew!!	NOT !!SK!! AND NOT !+desig+! AND NOT !!slew!!	!+desig+! AND NOT !!slew!!	!!slew!!
HUDdown1	!!SK!! AND NOT !+desig+! AND NOT !!slew!!	NOT !!SK!! AND NOT !+desig+! AND NOT !!slew!!	!+desig+! AND NOT !!slew!!	!!slew!!
SHUDdown1 *Snattack*	X	NOT !+desig+! AND NOT !!slew!!	!+desig+! AND NOT !!slew!!	!!slew!!
HUDdown2 *SHUDdown2*	X	NOT !+desig+! AND !+before slewing+! AND NOT !!slew!!	(!+desig+! OR NOT !+before slewing+!) AND NOT !!slew!!	!!slew!!
Noffset *Snoffset*	X	NOT !+desig+! AND !+before slewing+!	(!+desig+! AND NOT !!slew!!) OR (NOT !+desig+! AND !+after slewing+!)	!!slew!!
HUDUpd[a] *RadarUpd*	X	X	NOT !!slew!!	!!slew!!
A/A Guns *A/A Manrip* *A/G Guns*	X	X	NOT !!slew!!	!!slew!!
BOC *BOCFlyto0* *BOCoffset* *SBOC* *SBOCFlyto0* *SBOCoffset*	X	X	NOT !!slew!!	!!slew!!
Walleye	NOT !!slew!!	X	X	!!slew!!
!+AS elevation+!	!+boresight elevation+!	!+FPM elevation+!	!+HUDrefpt elev+!	!!Slewed AS elev!!
!+AS azimuth+!	!+boresight azimuth+!	!+FPM azimuth+!	!+HUDrefpt az+!	!!Slewed AS az!!

a. In this mode, the AS position is limited thus: Let *az* and *el* be the azimuth and elevation specified by the table. Then the limited azimuth := (*az* / |*az*|) X MIN(*az*, 5.5). The limited elevation is MIN(*el*, 4.3) if *el* is positive and MAX(*el*, -11.7) otherwise.

7.2.1 Local Dictionary

!!SK!! !+Weapon Class+! = SK

!!slew!! !+during slewing+!

!!Slewed AS az!!, !!Slewed AS elev!! These define the new position of the aiming symbol. Defined by Table 24.

TABLE 24. Values of !!Slewed AS az!! and !!Slewed AS elev!!

MODES	!!Slewed AS az!!, !!Slewed AS elev!!
RadarUpd *BOC* *BOCFlyto0* *BOCoffset* *SBOC* *SBOCFlyto0* *SBOCoffset*	!!Slewed AS az!! and !!Slewed AS elev!! are such that the HUD aiming symbol overlays the same point on the ground as as the FLR azimuth and range cursors.
HUDaln *Landaln* *I*, *OLB* *PolarI* *HUDUpd* *Nattack* *Noffset* *Snattack* *Snoffset*	!!Slewed AS az!! = !+AS azimuth+! + !+Slew HUD delta az+! !!Slewed AS elev!! = !+AS elevation+! + !+Slew HUD delta elev+!

7.3 Set HUD azimuth steering line (ASL) mode

TABLE 25. Set HUD azimuth steering line (ASL) mode

MODES	EVENTS		
BOC *BOCFlyto0* *BOCoffset* *CCIP* *HUDdown1* *HUDdown2* *Nattack* *Noffset* *SBOC* *SBOCflyto0* *SBOCoffset* *SHUDdown1* *SHUDdown2* *Snattack* *Snoffset*	@T(In mode)	@F(In Mode)	X
!+ASL mode+!	On	Off	$Intermittent$

7.4 Set the HUD azimuth steering line (ASL) position

TABLE 26. Set the HUD azimuth steering line (ASL) position.

MODES	CONDITIONS				
BOCFlyto0 *HUDdown1* *HUDdown2* *Nattack* *Noff-set* *SBOCFlyto0* *SHUDdown1* *SHUDdown2* *Snattack* *Snoffset*	X	NOT !+desig+!	!+steering to tgt+!	!+OTS+!	!+GAS+!
BOC *SBOC*	!+gr ac HUDrefpt+! > 30 nmi	X	!+gr ac HUDrefpt+! <= 30 nmi AND !+steer-ing to tgt+!	!+OTS+!	!+GAS+!
BOCoffset *SBOCoffset*	!+gr ac HUDrefpt+! > 30 nmi	NOT !+desig+! AND !+gr ac HUDrefpt+! <= 30 nmi	!+gr ac HUDrefpt+! <= 30 nmi AND !+steer-ing to tgt+!	!+OTS+!	!+GAS+!
!+ASL azimuth+!	!!error weight!! X !+steering error to tgt+!	!!AS intersection!!	!!near steering dis-play!!	-1/2 X !!error weight!! X !+steering error to tgt+!	!!closest edge!!

TABLE 27. Set the HUD azimuth steering line (ASL) position.

MODES	CONDITIONS	
CCIP	!+ip elev+! < -20	!+ip elev+! >= -20
!+ASL azimuth+!	!+FPM azimuth+!	!!ASL FPM intersect4 az!!
!+ ASL elevation+!	!+FPM elevation+!	4 degrees below !+FPM ele-vation+!

After !+ASL azimuth+! is computed according to the tables above, it is limited by the following formula:
(!+ALS azimuth+! / ABS(!+ALS azimuth+!)) X ABS(MIN(!+ASL azimuth+!, 6.7)) .

After that, ASL elevation is set to !!ASL elev placement!! in all cases.

TABLE 28. Set the HUD azimuth steering line (ASL) position

MODES	CONDITIONS		
BOC *BOCFlyto0* *BOCoffset* *HUDdown1* *HUDdown2* *Nattack* *Noffset* *SBOC* *SBOCFlyto0* *SBOCoffset* *SHUDdown1* *SHUDdown2* *Snattack* *Snoffset*	X	Always	X
CCIP	X	!+ip elev+! < -20	!+ip elev+! >= -20
!+ASL rotation+!	0	!+roll IMS+! X - 1	!+bomb fall line+!

7.4.1 Local Dictionary

!!alternate steering error!! !+steering error to tgt+! if !+gr ac tgt+! > 48,000 feet; !+steering error to rls+! otherwise.

!!AS intersection!! This is the azimuth angle that places the HUD ASL center on the imaginary line that runs through the FPM parallel to the pitch lines such that (given the current ASL rotation) the ASL will intersect the HUD aiming symbol.

!!ASL FPM intersect4 az!! The azimuth position of the ASL such that, given the current rotation angle of the ASL, (1) the ASL center is placed four degrees in elevation lower than the elevation of the Flight Path Marker; and (2) the ASL intersects the Flight Path Marker.

!!ASL elev placement!! The ASL elevation on the HUD such that, given the current ASL rotation and ASL azimuth, the ASL center is placed on the imaginary line that is both parallel to the pitch lines, and intersects the FPM. Limited to be no larger than 4.3 and no less than -11.7.

!!closest edge!! Edge of HUD to the closest return. If !!GAS left!! then #HUD symbol az min#; if NOT !!GAS left!! then #HUD symbol az max#.

!!error weight!! Defined by Table 29.

TABLE 29. Value of !!error weight!!

!+pitch system+! in degrees	!!error weight!!
!+pitch system+! <= 0	1
0 < !+pitch system+! <= 60	1 - 1.5 X !+pitch system+! / 360
60 < !+pitch system+! <= 80	3 - 13.5 X !+pitch system+! / 360
80 < !+pitch system+! <= 90	0

!!GAS left!! !+GAS+! AND (!+brg grtk tgt+! > 180).

!!near steering display!! If **NBnotShrike** or *SBOC* or *SBOCFlyto0* or *SBOCoffset* or *SHUDdown1* *SHUDdown2* or *Snattack* or *Snoffset* then 1/2 X !!error weight!! X !!alternate steering error!!
If **NBShrike** then !+AS azimuth+! + !+drift angle+!.

7.5 Set the HUD flight director mode

TABLE 30. Set the HUD flight director mode

MODES	EVENTS		
All alignment and navigation modes	@T(!+Init complete+!) OR @T(!+Weapon Mode+! = $None$)	@F(!+Weapon Mode+! = $None$)	X
!+FLTDIR mode+!	On	Off	$Intermittent$

7.6 Set HUD flight director azimuth position

TABLE 31. Set HUD flight director azimuth position

MODES	CONDITIONS	
All alignment and navigation modes	!+Fly to num+! != 0	!+Fly to num+! = 0
!+FLTDIR azimuth+!	!!ltd brg ac ftpt!!	0

7.6.1 Local Dictionary

!!ltd brg ac ftpt!! (!!steering error to ftpt!! / ABS(!!steering error to ftpt!!)) X MIN(!!steering error to ftpt!!, 5)

!!steering error to ftpt!! <!+brg ac ftpt+! modulo 360 - 360k, where k = 0 if !+brg ac ftpt+! modulo 360 <= 180 and k = 1 otherwise.

7.7 Set the HUD flight path marker (FPM) mode

TABLE 32. Set the HUD flight path marker (FPM) mode.

MODES	EVENTS		
No weapon mode listed below	@T(!+VV mode+!=On)	@F(!+VV mode+!=On)	X
A/A Manrip *BOC* *BOCFlyto0* *BOCoffset* *CCIP* *Manrip* *Nattack* *Noffset* *SBOC* *SBOCFlyto0* *SBOCoffset* *Snattack* *Snoffset* *Walleye*	@T(!+VV mode+!=On AND !!time FPM blinked!! >= 2.5 seconds)	@F(!+VV mode+!=On)	@T(!+VV mode+!=On AND !+stik empty+!)
!+FPM mode+!	On	Off	$Intermittent$

7.7.1 Local Dictionary

!!time FPM blinked!! Elapsed time since last occurrence of @T(!+FPM mode+! = $Intermittent$).

7.8 Set the HUD flight path marker (FPM) position

TABLE 33. Set the HUD flight path marker (FPM) position.

MODES	CONDITIONS		
Airaln	X	!+FM stage complete+! OR !+adc alt up+!	NOT !+FM stage complete+! AND NOT !+adc alt up+!
All alignment modes except *Airaln*	Always	X	X
DI *DIG* *PolarDI* *UDI*	X	Always	X
I *PolarI*	NOT !+in flight+!	!+in flight+!	X
Grid *OLB* *Mag sl*	NOT !+in flight+!	!+adc tas up+! AND !+in flight+!	NOT !+adc tas up+! AND !+in flight+!
IMS fail	NOT !+in flight+!	X	!+in flight+!
!+FPM elevation+!	0	!!ltd vert vels!!	!+AOA+!
!+FPM azimuth+!	0	!!ltd lat vels!!	0

7.8.1 Local Dictionary

!!ltd vert vels!! MIN(!!FPM elev from vels!!, 4.3) if !!FPM elev from vels!! > 0; MAX(!!FPM elev from vels!!, -11.7) if !!FPM elev from vels!! <= 0.

!!ltd lat vels! ! 0, if !!FPM az from vels!! = 0. Otherwise, ABS(!!FPM az from vels!!) / !!FPM az from vels!! X ABS(MIN((!!FPM az from vels!!, 6))

!!FPM az from vels!! The azimuth angle at which the FPM should be placed, assuming it is to depict the direction of the aircraft's velocity vector, derived from !!System velocities!!. Equal to the lateral component of system velocity divided by the forward component, where the result is interpreted as an angle in radians. The lateral and forward components are derived from !+velocity north system+! and !+velocity east system+!.

!!FPM elev from vels!! The elevation angle at which the FPM should be placed, assuming it is to depict the direction of the aircraft's velocity vector, derived from !!System velocities!!. Equal to the system vertical velocity divided by the forward component of the system velocity, where the result is interpreted as angle in radians. The forward component is derived from !+velocity north system+! and !+velocity east system+!.

!!System velocities!! The most recently calculated velocities from the most reliable available sensors.

7.9 Set the HUD in-range cue mode

TABLE 34. Set the HUD in-range cue mode

MODES	EVENTS		
A/G Guns	@T(!+target in range+!) OR @T(!+Gun Enable+! AND !+sr reasonable+!) WHEN(!+target in range+!)	@F(!+target in range+!) OR @F(In mode) OR @F(!+Gun Enable+! AND !+sr reasonable+!) WHEN(NOT !+target in range+!)	@F(!+sr reasonable+!) WHEN(!+Gun Enable+! AND !+target in range+!)
Walleye	@T(!+tgt ahead+! AND !+target in range+! AND !+desig+!)	@F(!+target in range+!) OR @F(!+desig+!) OR @F(In mode) OR @F(!+tgt ahead+!)	X
!+RNGCUE mode+!	On	Off	$Intermittent$

7.10 Set the HUD lower solution cue (LSC) mode

TABLE 35. Turning the LSC on and off.[a]

MODES	EVENTS	
NBnotShrike	@T(!+target in range+! AND !+desig+! AND NOT !+GAS+! AND NOT !+during slewing+!) OR @T(1 sec before !+target in range+!) WHEN(!+pitch IMS+!=42)	@F(!+target in range+! AND !+desig+! AND NOT !+GAS+! AND NOT !+during slewing+!)
NBShrike	@T(!+target in range+! AND !+desig+!)	@F(!+target in range+! AND !+desig+!)
CCIP	@T(!!impact angle proper!!)	@F(!!impact angle proper!!)
SBOC *SBOCflyto0* *SBOCoffset* *SHUDdown1* *SHUDdown2* *Snattack* *Snoffset*	@T((!+special in range+! AND !+desig+! AND NOT !+GAS+! AND NOT !+during slewing+!) AND (NOT !+low drag release+! OR !+tgt ahead+!))	@T((NOT !+special in range+! OR NOT In mode OR !+GAS+! OR !+during slewing+!) AND (NOT !+low drag release+! OR !+OTS+! OR !+rmax+! OR NOT !+tgt ahead+!))
!*None*!	X	@T(In mode)
!+LSC mode+!	On	Off

a. This table only applies when the LSC mode is not $Intermittent$. That is, the table that sets the mode to $Intermittent$ takes precedence.

TABLE 36. Flashing the LSC.

MODES	EVENTS	
HUDdown1 *HUDdown2* *Nattack* *Noffset* *SHUDdown1* *SHUDdown2* *Snattack* *Snoffset*	@T(!+desig+! AND NOT !+sr reasonable+!) OR @F(!+Slew displacement non-zero+! AND NOT !+sr reasonable+!) WHEN(!+desig+!)	@T(!+desig+! AND !+sr reasonable+!) OR @F(!+Slew displacement non-zero+! AND !+sr reasonable+!) WHEN(!+desig+!)
None of the modes above and not in *Grtest*	X	@T(In mode)
!+LSC mode+!	$Intermittent$	!!stale LSC mode!!

7.10.1 Local Dictionary

!!impact angle proper!! ABS(!+ip elev+!) <= 16 AND ABS(!+ip az+!) <= 12)

!*None*! For purposes of this function, the system is in mode !*None*! when it is not in any of the following modes: **NBShrike**, **NBnotShrike**, *CCIP*, *Snattack*, *Snoffset*, *SBOC*, *SBOCFlyto0*, *SBOCoffset*, *SHUDdown1*, OR *SHUDdown2*.

!!stale LSC mode!! The value of !+LSC mode+! as determined by the table to turn the LSC on and off.

7.11 Set the HUD lower solution cue (LSC) position

TABLE 37. Set the HUD lower solution cue (LSC) position.

MODES	CONDITIONS		
BOC *BOCflyto0* *BOCoffset* *HUDdown1* *HUDdown2* *Nattack* *Noffset*	!+OTS+!	!+low drag release+! AND NOT !+OTS+!	NOT !+low drag release+! AND NOT !+OTS+!
!+LSC elevation+!	!+ASL elevation+! - 4	!+ASL elevation+! - !!ltd dive pullup!!	!+ASL elevation+! - !!ltd sr ac rls!!
!+LSC azimuth+!	!!LSC az on ASL!!	!!LSC az on ASL!!	!!LSC az on ASL!!

TABLE 38. Set the HUD lower solution cue (LSC) position.

MODES	CONDITIONS		
HiNuke	!+TOS+!	NOT !+TOS+! AND !+sr ac rls+! >= 0 ft	NOT !+TOS+! AND !+sr ac rls+! < 0 ft
!+LSC elevation+!	!+FPM elevation+! + 4	!+FPM elevation+! + !!ltd sr ac rls!!	!+FPM elevation+! + !!wtd sr ac rls!!
!+LSC azimuth+!	!!LSC az on ASL!!	!!LSC az on ASL!!	!!LSC az on ASL!!

TABLE 39. Set the HUD lower solution cue (LSC) position.

MODE	!+LSC elevation+!	!+LSC azimuth+!
CCIP	!+ip elev+!	!!LSC az on ASL!!
LoNuke	!+FPM elevation+! + !!wtd gracrmax!!	!!LSC az on ASL!!

7.11.1 Local Dictionary

!!LSC az on ASL!! !+LSC az on ASL+!; this is the azimuth angle at which to place the LSC so that it intersects the ASL.

!!ltd dive pullup!! +(!+dive pullup+! / ABS(!+dive pullup+!)) X MIN(4, 1/8, ABS(!+dive pullup+!)), where !+dive pullup+! is converted to a real in degrees.

!!ltd sr ac rls!! +(!+sr ac rls+! / ABS(!+sr ac rls+!)) X MIN(4, .001, ABS(!+sr ac rls+!)), where !+sr ac rls+! is converted to a real in feet.

!!wtd gracrmax!! MIN(1, 4, .001, ABS(!+gr ac rmax+!)), where !+gr ac rmax+! is converted to a real in feet.

!!wtd sr ac rls! -1 X MIN(3.5, .001, ABS(!+sr ac rls+!)), where !+sr ac rls+! is converted to a real in feet.

7.12 Set the HUD pullup anticipation cue (PUAC) mode

TABLE 40. Set the HUD pullup anticiation cue (PUAC) mode.

MODES	EVENTS		
BOCFlyto0 *CCIP* *Nat-tack* *Noffset*	@T(In mode AND !+Master Arm+!)	X	@T(In mode AND NOT !+Master Arm+!)
A/G Guns	@T(In mode AND (!+Master Arm+! OR !+Weapon Class+! = GN OR RK)	X	@T(In mode AND NOT !+Master Arm+!) WHEN(!+Weapon Class+! = GN OR RK)
BOC *BOCoffset*	@T(In mode AND !+Master Arm+! AND !+gr ac ftpt+! <= 30 nmi)	@T(!+gr ac ftpt+! > 30 nmi AND !+Master Arm+!)	@T(In mode AND NOT !+Master Arm+!)
*SBOC *SBOCFlyto0* *SBOCoffset* *Snattack* *Snoffset*	@T((!+Master Arm+! OR !+high drag release+! OR !+rmax+6000+!)	@T(!+Master Arm+! AND !!Off special!!) OR @T(!+high drag release+!)	@T(In mode AND NOT !+Master Arm+!) WHEN(!+low drag release+!)
Walleye	@T(In mode AND !+Master Arm+!)	X	@T(In mode AND NOT !+Master Arm+!)
No other listed modes	X	@T(!+Weapon Mode+! = $None$)	X
!+PUAC mode+!	On	Off	$Intermittent$

7.12.1 Local Dictionary

!!Off special!! (!+rmin+6000+! AND !+stik created+!) OR (!+gr ac ftpt+!>= 10 nmi AND !+stik empty+!) OR (!+pitch IMS+! < -30 AND !+stik empty+!)

7.13 Set the HUD pullup anticipation cue (PUAC) position

TABLE 41. Setting PUAC elevation.

MODES	CONDITIONS	
BOC *BOCFlyto0* *BOCoffset* *CCIP* *Nattack* *Noffset* *Walleye*	!+sr ac gpup+! > 5000 ft AND !+sr ac btpup+! > 5000 ft	!+sr ac gpup+! <= 5000 ft OR !+sr ac btpup+! <= 5000 ft
A/G Guns	!+sr ac gpup+! > 5000 ft AND (!+sr ac btpup+! > 5000 ft OR !+Weapon Class+! = GN)	!+sr ac gpup+! <= 5000 ft OR (!+Weapon Class+! = RK AND !+sr ac btpup+! <= 5000 ft)
SBOC *SBOCFlyto0* *SBOCoffset* *Snattack* *Snoffset* *SHUDdown1* *SHUDdown2*	@T(in mode)	X
!+PUAC elevation+!	!+FPM elevation+! - 3.5	!!pullup elev!!

For the PUAC elevation requirements in *Snattack*, *Snoffset*, *SBOC*, *SBOCFlyto0*, and *SBOCoffset*modes, see the classified Addendum.

TABLE 42. Setting HUD PUAC azimuth.

MODES	!+PUAC azimuth+!
BOC *BOCFlyto0* *BOCoffset* *Nattack* *Noffset* *SBOC* *SBOCFlyto0* *SBOCoffset* *Snattack* *Snoffset*	!!PUAC az on ASL!!
A/G Guns *CCIP* *Walleye*	!+FPM azimuth+!

7.13.1 Local Dictionary

!!pullup elev!! !+FPM elevation+! - (0.7 X MIN(!+sr ac btpup+!, !+sr ac gpup+!) / 1000 feet)

!!PUAC az on ASL!! !+PUAC az on ASL+!; this is the azimuth angle at which to place the PUAC so that it intersects the ASL.

7.14 Set the HUD pullup cue mode

TABLE 43. Set the HUD pullup cue mode

MODES	EVENTS		
All modes except *Grtest*	X	@T(!+blast danger+! OR !+ground danger+!)	@F(!+blast danger+! OR !+ground danger+!) OR @T(!!time PUC blinked!! >= 2 sec) WHEN(!+low drag release+! AND !!Special!!)
!+PUC mode+!	On	$Intermittent$	Off

7.14.1 Local Dictionary

!!Special!! !+Weapon Class+! = SOD OR SSH

!!time PUC blinked!! Elapsed time since last occurrence of @T(!+PUC mode+! = $Intermittent$)

7.15 Set the HUD upper solution cue (USC) mode

TABLE 44. Turning the USC on and off.[a]

MODES	EVENTS	
NBnotShrike *SBOC* *SBOCFlyto0* *SBOCoffset* *SHUDdown1* *SHUDdown2* *Snattack* *Snoffset*	@T(!+special in range+! AND !+desig+! AND NOT !+GAS+! AND NOT !+during slewing+!) WHEN(!+low drag release+!)	@F(In mode) OR @F(!+special in range+! AND NOT !+GAS+! AND NOT !+during slewing+!) OR @F(!+low drag release+!)
!+USC mode+!	On	Off

a. This table only applies when the USC mode is not $Intermittent$. That is, the table that sets the mode to $Intermittent$ takes precedence.

TABLE 45. Flashing the USC.

MODES	EVENTS	
HUDdown1 *HUDdown2* *Nattack* *Noffset* *SHUDdown1* *SHUDdown2* *Snattack* *Snoffset*	@T(!+LSC mode+! = $Intermittent$)	@F(!+LSC mode+! = $Intermittent$)
No mode listed above and not *Grtest*	X	@T(In mode)
!+USC mode+!	$Intermittent$	!!stale USC mode!!

7.15.1 Local Dictionary

!!stale USC mode!! The value of !+USC mode+! as determined by the table to turn the USC on and off.

7.16 Set the HUD upper solution cue (USC) position

TABLE 46. Set the HUD upper solution cue (USC) position.

MODES	CONDITIONS	
NBnotShrike *SBOC* *SBOCFlyto0* *SBOCoff-set* *SHUDdown1* *SHUDdown2* *Snattack* *Snoffset*	!+OTS+!	NOT !+OTS+!
!+USC elevation+!	!!ltd OTS pullup!!	!!ltd loft pullup!!
!+USC azimuth+!	!!USC az on ASL!!	!!USC az on ASL!!

7.16.1 Local Dictionary

!!ltd loft pullup!! (!+loft pullup+! / ABS(!+loft pullup+!)) X MIN(4, 1/8, ABS(!+loft pullup+!)), where !+loft pullup+! is converted to a real in degrees.

!!ltd OTS pullup!! (!+OTS pullup+! / ABS(!+OTS pullup+!)) X MIN(4, 1/8, ABS(!+OTS pullup+!)), where !+OTS pullup+! is converted to a real in degrees.

!!USC az on ASL!! !+USC az on ASL+!; this is the azimuth angle at which to place the USC so that it intersects the ASL.

7.17 Set the HUD symbol blink period

This function is never performed, because under the current requirements, the blink period for all HUD symbols is the same, and is equal to the default rate.

7.18 Display the HUD test patterns

TABLE 47. Display the HUD test patterns

MODES	EVENTS		
Grtest	@T(!+test stage+! =SC) WHEN !+Weapon Mode+! = $None$)	@T(!+test stage+! =$SC) WHEN !+Weapon Mode+! != $None$)	@F(!+test stage+! = SC)
!+HUD test mode+!	A	B	$None$

8.0 Head-Up Display Value-Indicator Functions

8.1 Set the HUD altitude display

TABLE 48. Set the HUD altitude display.

MODES	CONDITIONS		
All alignment and navigation modes	!+adc alt up+!	NOT !+adc alt up+!	X
Grtest	X	X	Always
!+HUD alt+!	!+alt ADC+!	4500 ft	!!last pre-test value!!

8.1.1 Local Dictionary

!!last pre-test value!! The value that was being output when @T(*Grtest*) occurred.

8.2 Set the HUD heading display

TABLE 49. Set the HUD heading display.

MODES	CONDITIONS	
All alignment and navigation modes, except *IMS fail*	Always	X
IMS fail	!+IMS mode+! != Off	!+IMS mode+! = Off
Grtest	X	Always
!+HUD heading+!	!+heading MAG+!	0 (North)

8.3 Set the HUD pitch display

TABLE 50. Set the HUD pitch display.

MODES	!+HUD pitch+!	!+HUD roll+!
All navigation and alignment modes	!+pitch system+!	!+roll system+!
Grtest	!!pre-test pitch!!	!!pre-test roll!!

8.3.1 Local Dictionary

!!pre-test pitch!! The value of !+pitch system+! when @T(*Grtest*) occurred.

!!pre-test roll!! The value of !+roll system+! when @T(*Grtest*) occurred.

8.4 Enable the HUD vertical velocity and vertical acceleration displays

TABLE 51. Enable the HUD vertical velocity and vertical acceleration displays.

MODES	EVENTS		
Airaln	@F(!+align stage+!=FM OR @T(!+AOA valid+!)	@F(!+AOA valid+!) WHEN(NOT !+FM stage complete+!)	X
IMS fail *Grid* *Mag sl* *OLB*	@T(!+AOA valid+!) OR @F(!+in flight+!)	@T(!+in flight+! AND NOT !+AOA valid+!)	X
No other listed mode	@T(In mode)	X	X
!+VV mode+!	On	Off	$Intermittent$

8.5 Set the HUD vertical acceleration display

8.5.1 Initiation/Termination Events

TABLE 52. Set the HUD vertical acceleration display.

MODES	Initiation events	Termination events
Snattack *Snoffset* *SBOC* *SBOCFlyto0* *SBOCoffset* *SHUDdown1* *SHUDdown2*	@T(In mode AND !+low drag release+! AND !+VV mode+! = On)	@F(!+low drag release+! AND !+VV mode+! = On) OR @F(In mode)

8.5.2 Function Definition

!+HUD NACC+! = !+normal accel+!

8.6 Set the HUD vertical velocity display

8.6.1 Initiation/Termination Events

Initiation: When the function to set the HUD vertical acceleration terminates, provided it was not terminated by !+VV mode+! becoming Off, OR @T(!+Init complete+!).

Termination: When the function to set the HUD vertical acceleration initiates, or @T(!+VV mode+!) = Off.

8.6.2 Function Definition

TABLE 53. Set the HUD vertical velocity display.

MODES	CONDITIONS	
All alignment modes *DIG* *DI* *I* *PolarDI* *PolarI* *UDI* *Grid* *IMS fail* *Mag Sl* *OLB*	Always	X
Grtest	X	Always
!+HUD vertvel+!	!+velocity vertical system+!	0 fps

9.0 Inertial Measurement Set Functions

9.1 Turn the computer control of the IMS on/off

TABLE 54. Turn computer control of the IMS on/off.

MODES	EVENTS	
Airaln	@T(!!roll lrg!! AND !+align stage+!=FM) OR @T(In mode AND !+align stage+!\(!=FM)	@F(!!roll lrg!!) WHEN (!+align stage+! = FM) OR @T(!+align stage+! = FM) WHEN(NOT !!roll lrg!!) OR @T(!+align stage+! != CL AND !+IMS reasonable+!) WHEN(!+in flight+!)
Any align- ment mode but *Airaln*	@T(In mode)	@T(!+align stage+! != CL AND !+IMS reasonable+!) WHEN(!+in flight+!)
DIG *DI* *I* *OLB* *PolarDI* *PolarI* *UDI*	@T(In mode)	X
Grid *IMS fail* *Mag Sl*	X	@T(In mode)
!+IMS enable+!	true	false

9.1.1 Local Dictionary

!!roll lrg!! ABS(!+roll IMS+!) > 5.

9.2 Set the IMS velocity measurement scale

TABLE 55. Set the IMS velocity measurement scale.

MODES	EVENTS	
Landaln *Lautocal* *01 Update*	@T(In mode)	X
HUDaln	@T(In mode) WHEN (!+IMS mode+! = $Gndal$)	@T(In mode) WHEN(!+IMS mode+! = ($Norm$ OR $Iner$))
Airaln *Sautocal* *SIN-Saln* *DI* *DIG* *I* *OLB* *PolarDI* *PolarI* *UDI*	X	@T(In mode)
!+IMS scale+!	$Fine$	$Coarse$

9.3 Make small adjustments to platform X axis

TABLE 56. Make small adjustments to platform X axis.

MODES	CONDITIONS
Lautocal	!+align stage eq FG+! OR !+align stage eq ND+! OR !+align stage eq ND2+! OR !+align stage eq ED+! OR !+align stage eq ED2+!
Sautocal	!+align stage eq ED+! OR !+align stage eq ED2+! OR !+align stage eq ND+! OR !+align stage eq ND2+!
01Update *HUDaln* *SINSaln*	!+align stage eq FG+!
Airaln	!+align stage eq HL+! OR !+align stage eq FG+! OR !+align stage eq HG+!
DI *DIG* *I* *OLB* *PolarDI* *PolarI* *UDI* *Landaln*	Always
!+X fine rotation+!	!!IMS adj x error!!
!+Y fine rotation+!	!!IMS adj y error!!

9.3.1 Local Dictionary

!!IMS adj x error!!, !!IMS adj y error!! Defined by the Table 57.

TABLE 57. Values of x and y adjustments: !!IMS adj x error!! and !!IMS adj y error!!

MODES	!!IMS adj x error!!	!!IMS adj y error!!
Landaln *Lautocal* *HUDaln* *01Update*	!!ims x const error mc!!	!!ims y const error mc!!
Sautocal *SINSaln*	!!ims x sins error mc!!	!!ims y sins error mc!!
Airaln *DIG* *DI* *PolarDI*	!!ims x dop error mc!!	!!ims y dop error mc!!
I *UDI* *OLB* *PolarI*	!!ims x nav error m!!	!!ims y nav error m!!

!!ims x const error mc!!, !!ims y const error mc!! The angular adjustment required to maintain and correct the IMS x and y alignment, respectively, assuming that the aircraft is not moving. The correction is applied during the $CA2$, $CL2$ and $FG2$ stages, and the maintenance is applied periodically during the mode.

!!ims x sins error mc!!, !!ims y sins error mc!! The angular adjustment required to maintain and correct the IMS x and y axis alignment,respectively, assuming that the aircraft is moving as indicated by SINS inputs.

!!ims x dop error mc!!, !!ims y dop error mc!! The angular adjustment required to maintain and correct the IMS x and y axis alignment, respectively, assuming that the aircraft is moving as indicated by Doppler inputs.

!!ims x nav error m!!, !!ims y nav error m!! The angular adjustment required to maintain and correct the IMS x and y axis alignment, respectively, assuming that the aircraft is moving as indicated by IMS inputs.

9.4 Perform large adjustments of the IMS platform x axis

TABLE 58. Perform large adjustments of the IMS platform x axis.

MODES	EVENTS
HUDaln *Lautocal* *Landaln* *Sautocal* *SINSaln*	@T(!+align stage eq CL2+! OR !+align stage eq CA2+!)
Airaln	@T(!!IMS adj xy error!! > #IMS adj xy tolerance#) WHEN(!+align stage eq CL2+!)
!+X coarse rotation+!	!!IMS adj x error!!
!+Y coarse rotation+!	!!IMS adj y error!!

9.4.1 Local Dictionary

!!IMS adj xy error!! MAX(!!IMS adj x error!!, !!IMS adj y error!!)

!!IMS adj x error!!, !!IMS adj y error!! Defined by Table 57.

9.5 Fine adjustments to alignment of IMS z axis

TABLE 59. Fine adjustments to alignment of IMS z axis.

MODES	CONDITIONS
Lautocal *Sautocal*	!+align stage eq CA2+! OR !+align stage eq ND2+! OR !+align stage eq ED2+! OR !+align stage eq FG+!
HUDaln	!+align stage eq CA2+! OR !+align stage eq FG+!
Airaln	!+align stage eq HL+! OR !+align stage eq FG+!
01Update	!+align stage eq FG+!
DI *DIG* *Landaln*	Always
I *OLB* *UDI*	!+latitude+! <= 80
!+Z fine rotation+!	!!IMS small z error!!

9.5.1 Local Dictionary

!!IMS small z error!! The amount of small correction to be applied to the IMS z axis; values defined by Table 60.

TABLE 60. Value of !!Slewed AS az!!, !!Slewed AS elev!!

MODES	!!IMS small z error!!
HUDaln *Landaln* *Lautocal* *01Update*	!!ims z const error mc!!
Sautocal *SINSaln*	!!ims z sins error mc!!
Airaln *DIG*	!!ims z dop error mc!!
DI	!!ims z dop error m!!
I *OLB* *UDI*	!!ims z nav error m!!

!!ims z const error mc!! The angular adjustment to apply to the IMS z axis to maintain and correct alignment,assuming the aircraft is not moving. The adjustment to correct alignment in the IMSalignment are only applied in $FG2$ and $CA2$ stages, but the maintenance adjustments are applied periodically throughout the mode.

!!ims z sins error mc!! The angular adjustment to apply to the IMS z axis to maintain and correct alignment, assuming the aircraft is moving as indicated by SINS inputs.

!!ims z dop error mc!! The angular adjustment to apply to the IMS z axis to maintain and correct alignment, assuming the aircraft is moving as indicated by Doppler inputs.

!!ims z dop error m!! The angular adjustment to apply to the IMS z axis to maintain alignment, assuming the aircraft is moving as indicated by Doppler inputs.

!!ims z nav error m!! The angular adjustment to apply to the IMS z axis to maintain alignment, assuming the aircraft is moving as indicated by IMS inputs.

9.6 Coarse adjustments to alignment of IMS z axis

TABLE 61. Coarse adjustments to alignment of IMS z axis.

MODES	EVENTS	
HUDaln	@T(!+TD pressed+!) WHEN (NOT !+IMS coarse rotating+!)	@T(!+align stage eq CA2+!) WHEN(!!IMS subsequent z adj!! > #ims cutoff#)
Lautocal	X	@T(!+align stage eq CA2+! OR !+align stage eq ND2+! OR !+align stage eq ED2+!) WHEN(!!IMS subsequent z adj!! > #ims cutoff#)
Landaln	X	@T(!+align stage eq CA2+! OR !+align stage eq FG2+!) WHEN(!!IMS subsequent z adj!! > #ims cutoff#)
SINSaln	@T(In mode)	@T(!+align stage eq CA2+!) WHEN(!!IMS subsequent z adj!! > #ims cutoff#)
!+Z coarse rotation+!	!!IMS preliminary z adj!!	!!IMS subsequent z adj!!

TABLE 62. Coarse adjustments to alignment of IMS z axis.

MODES	EVENTS		
Lautocal	X	@F(!+align stage eq ED+! OR !+align stage eq ED2+!)	@T(!+align stage eq ED+! OR !+align stage eq ED2+!)
Sautocal	@F(!+align stage eq ED+! OR !+align stage eq ED2+!)	X	@T(!+align stage eq ED+! OR !+align stage eq ED2+!)
!+Z coarse rotation+!	90 deg CCW + !!ims z const error mc!!	90 deg CCW + !!ims z sins error mc!!	90 deg CW

9.6.1 Local Dictionary

!!az ref error!! The error (difference in heading) calculated by comparing the IMS heading to the heading calculated from !+az ref hdg pnl+! corrected by the HUD aiming symbol azimuth displacement at the time of the test. The value is updated whenever @T(!+desig+!) WHEN(!+in hudaln+!) occurs. The value is (!+az ref hdg pnl+! + !+AS azimuth+! - !+heading IMS+!) modulo 360.

!!sins error!! Angular difference measured from !+heading IMS+! to (!+SINS heading+! + !+SINS dhdg pnl+!). Positive if that angle is measured clockwise; negative if counterclockwise.

!!IMS preliminary z adj!!, !!IMS subsequent z adj!! The amount of large (coarse) correction to be applied to the IMS z axis during either a preliminary adjustment or a subsequent adjustment, respectively. Values defined in Table 63.

TABLE 63. Values of preliminary and subsequent IMS z-axis adjustments.

MODES	!!IMS preliminary z adj!!	!!IMS subsequent z adj!!
HUDaln	!!az ref error!!	!!ims z const error mc!!
SINSaln	!!sins error!!	!!ims z sins error mc!!
Landaln	X	!!ims z const error mc!!

!!ims z const error mc!! Defined in previous section.

!!ims z sins error mc!! Defined in previous section.

!!ims z dop error mc!! Defined in previous section.

!! ims z dop error m!! Defined in previous section.

!!ims z nav error m!! Defined in previous section.

9.7 Initialize the IMS horizontal velocities

TABLE 64. Initialize the IMS horizontal velocities.

MODES	EVENTS	
SINSaln *Sautocal*	@T(!+new align stage+!)	X
Landaln	X	@T(!+align stage eq CL+!) OR @T(!+align stage eq CA+!) OR @T(!+align stage eq TS+!) OR (@T(!+align stage eq FG+!) WHEN (!+land velocity test passed+! OR (NOT !+TS stage complete+!))
Lautocal	X	@T(In mode)
IMS fail	X	@T(In mode)
!+IMS E velocity+!	!+SINS east vel+!	0 fps
!+IMS N velocity+!	!+SINS north vel+!	0 fps

9.8 Initialize the IMS vertical velocity

TABLE 65. Initialize the IMS vertical velocity.

MODES	EVENTS
All modes	@T(!+power up+!) OR @F(!+IMS up+!)
!+IMS V velocity+!	0 fps

9.9 Set the IMS reconfiguration values

TABLE 66. The value of !+X drift+!

MODES	EVENTS
All modes except *Grtest*	@T(!+new Xdrift pnl entered+!)
!+X drift+!	!+X drift pnl+!

TABLE 67. The value of !+Y drift+!

MODES	EVENTS
All modes except *Grtest*	@T(!+new Y drift pnl entered+!)
!+Y drift+!	!+Y drift pnl+!

TABLE 68. The value of !+Z drift +!

MODES	EVENTS
All modes except *Grtest*	@T(!+new Z drift pnl entered+!)
!+Z drift +!	!+Z drift pnl+!

TABLE 69. The value of !+X corr increm+!

MODES	EVENTS
All modes except *Grtest*	@T(!+new X corr increm pnl entered+!)
!+X corr increm+!	!+X corr increm pnl+!

TABLE 70. The value of !+Y corr increm+!

MODES	EVENTS
All modes except *Grtest*	@T(!+new Y corr increm pnl entered+!)
!+Y corr increm+!	!+Y corr increm pnl+!

TABLE 71. The value of !+Z corr increm+!

MODES	EVENTS
All modes except *Grtest*	@T(!+new Z corr increm pnl entered+!)
!+Z corr increm+!	!+Z corr increm pnl+!

TABLE 72. The value of !+N coarse scale+!

MODES	EVENTS
All modes except *Grtest*	@T(!+new N coarse scale pnl entered+!)
!+N coarse scale+!	!+N coarse scale pnl+!

TABLE 73. The value of !+E coarse scale+!

MODES	EVENTS
All modes except *Grtest*	@T(!+new !E coarse scale pnl entered+!)
!+E coarse scale+!	!+E coarse scale pnl+!

TABLE 74. The value of !+V coarse scale+!

MODES	EVENTS
All modes except *Grtest*	@T(!+new V coarse scale pnl entered+!)
!+V coarse scale+!	!+V coarse scale pnl+!

TABLE 75. The value of !+N fine scale+!

MODES	EVENTS
All modes except *Grtest*	@T(!+new N fine scale pnl entered+!)
!+N fine scale+!	!+N fine scale pnl+!

TABLE 76. The value of !+E fine scale+!

MODES	EVENTS
All modes except *Grtest*	@T(!+new E fine scale pnl entered+!)
!+E fine scale+!	!+E fine scale pnl+!

TABLE 77. The value of !+N coarse bias+!

MODES	EVENTS
All modes except *Grtest*	@T(!+new N coarse bias pnl entered+!)
!+N coarse bias+!	!+N coarse bias pnl +!

TABLE 78. The value of !+E coarse bias+!

MODES	EVENTS
All modes except *Grtest*	@T(!+new E coarse bias pnl entered+!)
!+E coarse bias+!	!+E coarse bias pnl +!

TABLE 79. The value of !+V coarse bias+!

MODES	EVENTS
All modes except *Grtest*	@T(!+new V coarse bias pnl entered+!)
!+V coarse bias+!	!+V coarse bias pnl +!

TABLE 80. The value of !+N fine bias+!

MODES	EVENTS
All modes except *Grtest*	@T(!+new N fine bias pnl entered+!)
!+N fine bias+!	!+N fine bias pnl +!

TABLE 81. The value of !+E fine bias+!

MODES	EVENTS
All modes except *Grtest*	@T(!+new E fine bias pnl entered+!)
!+E fine bias+!	!+E fine bias pnl +!

10.0 Panel Functions

10.1 Set the panel's mark window display

TABLE 82. Set the panel's mark window display.

MODES	EVENT
All modes except *Grtest*	@T(!+Init complete+!)
Grtest	@T(In mode)
!+Mark window+!	character ' '

TABLE 83. Set the panel's mark window display.

MODES	EVENTS		
All modes except *Grtest*	@T(!+mark pressed+!) WHEN(NOT !!Nav2 config!!)	@T(!!Nav2 config!! AND NOT !+adc reasonable+!)	@T(!!Nav2 config!! AND !+adc reasonable+!)
!+Mark window+!	!+mark+!, converted to character form	character '0'	character '1'

10.1.1 Local Dictionary

!!Nav2 config!! True iff !+pnl config+! = $nav diags2$.

10.2 Control the panel's enter light

TABLE 84. Control the panel's enter light.

MODES	EVENTS	
FlyUpd *HUDUpd* *MapUpd* *RadarUpd* *TacUpd*	@T(!+TD pressed+!) WHEN(NOT !+Enter light+!) OR @T(!+data enterable+!)	!!Panel switch changed!! OR @T(!+input attempted+!) OR @T(!+Fly to num changed+!) OR @T(!+Fly to state changed+!)) WHEN(!+Enter light+!) OR @T(!+Init complete+!) OR @F(!+data enterable+!)
!+Enter light+!	true	false

10.2.1 Local Dictionary

!!Panel switch changed!! @T(!+Panel mode changed+!) OR @T(!+Update changed+!) OR @T(!+Pres pos changed+!) OR @T(!+Map hold changed+!) OR @T(!+Enter pressed+!) OR @T(!+input requested+!).

10.3 Display data in the upper or lower window

10.3.1 What to display on the panel windows

Table 85 and Table 86 show:

1. What should be displayed in the upper and lower panel windows, respectively.
2. The value of !+panel config+! in the leftmost column in each table determines what is displayed in each window (rightmost column) and the format of the display (in the adjacent column).
3. The Input Type columns lists the data type of the !!terms!! whose types are not provided in their definitions in the local dictionary.
4. The Output Type columns list the type that the item must have in order to be displayed (if that data type is different from that item's input data type). If different, a type conversion will be required. In most cases, the value mapping from one type to another is obvious (e.g., from "integer" to "real"), and will not be discussed. In cases where the mapping is not obvious (e.g., from "angle" to "boolean") it will be given in the "Notes" section following the table.
5. If the produced display value exceeds the constraints listed in the Value Constraints column, this function shall display the nearest limiting value.

10.3.2 When to display an !!item!!

The appropriate values are displayed in the windows each time the following occurs: @T(!+pnl config changed+!) A value is re-displayed every time its value changes by at least !!resolution!! amount, until @T(!+pnl config changed+!) next occurs. The term !!resolution!! is defined for each item in the local dictionary.

10.3.3 When to display nothing

The functions to update displays shall also cease when @T(!+input attempted+!), @T(!+input requested+!), or @T(!+panel error+!) occurs, and shall resume when @T(!+pnl input complete+!) or @F(!+panel error+!) occurs.

10.3.4 The panel display in *Grtest*

When @T(*Grtest*) occurs, both windows shall continue to display what they did at mode entry. When @T(!+test stage+! = PD) occurs, all format lights shall be turned on and the windows shall display all "8"s. No other panel display will occur as long as the system is in *Grtest* mode. When the mode is exited, the display corresponding to the current !+pnl config+! resumes.

TABLE 85. Display in Upper Panel Window

!+pnl config+!	Input Type	Output type (if different)	Value Constraints	!+up win fmt+!	!+up win val+!
$align stage$	astage	char_string		CHARSTR	!!align stage!!
$alt baro AGL$	distance	integer	0 - 65535 feet	UINT	!!alt baro AGL!!
$ARPQUANT$	integer		0 - 99	UINT	!!Weap Quantity!!
$az miss dist at rls$	distance	integer	0 - 65535 ft	UINT	!!az miss dist at rls!!
$data nbr$	integer		0 - 26	BINT	!+data nbr pnl+!
$dest altitude$	distance	integer	-65535 - 65535ft	SINT	!+dest altitude pnl+!(i)
$dest lat$	latitude		N90-S90	LATITUDE	!+dest lat+! (i)
$drftangl IMS$	angle	integer	-180 - +180	SINT	!!drftangl IMS!!
$groundspeed IMS$	speed	integer	0 - 1214 knots	UINT	!!groundspeed IMS!!
$hdg system$	angle		0 - 360	ANGLE	!+hdg system+!
$heading IMS$	angle		0 - 360	ANGLE	!+heading IMS+!
$IMS diags1$	various	char_string		CHARSTR	!!IMS diags1!!
$IMS total vel$	speed	integer	0 - 1214 knots	UINT	!!IMS total vel!!
$latitude error$	latitude		N90-S90	LATITUDE	!+latitude error+!
$latitude$	latitude		N90-S90	LATITUDE	!+latitude+!
$mag variation$	--	--	--	--	!!North light!!
$map latitude$	latitude		N90-S90	LATITUDE	!+Map latitude+!
$mark lat$	latitude		N90-S90	LATITUDE	!+mark lat+! (i)
$MFSW diags$	various	char_string		CHARSTR	!!MFSW diags!!
$nav diags1$	various	char_string		CHARSTR	!!Nav diags1!!
$none$	--	--	--	--	!!blanks!!
$offset dht$	distance	integer	-65535 - 65535ft	SINT	!+offset dht pnl+! (i)
$offset rng$	distance	integer	0-131,070 ft	UINT	!+offset rng pnl+! (i)
$OFP ver1$	char_string	--	--	CHARSTR	!!OFP version upper!!
$SINS lat$	latitude		N90-S90	LATITUDE	!+SINS lat+!
$SINS valid1$	various	char_string		CHARSTR	!!SINS valid1!!
$SINS x offset$	distance	integer	-2047 - +2047 ft	SINT	!+SINS x offset pnl+!
$SINS z offset$	distance	integer	-2047 - +2047 ft	SINT	!+SINS z offset pnl+!
$slant range at rls$	distance	integer	0 - 262141 ft	UINT	!!Slant range at rls!!
$STARDY diags$	various	char_string		CHARSTR	!!STARDY diags!!
$TAS ADC at rls$	speed	integer	0 - 32767 knots	UINT	!!TAS ADC at rls!!
$vel e$	speed	integer	-2047 - +2047kts	SINT	!!vel E!!
$vel n$	speed	integer	-2047 - +2047kts	SINT	!!vel N!!
$wind speed$	speed	integer	0 - 255 knots	UINT	!!wind speed!!

TABLE 86. Display in Lower Panel Window (Sheet 1 of 3)

!+pnl config+!	Input Type	Output type (if different)	Value Constraints	!+low win fmt+!	!+low win val+!
$alt AGL at rls$	distance	integer	0 - 65535 ft	UINT	!!Alt AGL at rls!!
$ARPINT$	integer		0 - 990	UINT	!!Weap Interval!!
$az ref hdg$	angle		0 - 360	ANGLE	!+az ref hdg pnl+!
$burst ht$	distance	integer	0-65535 ft	UINT	!+burst ht pnl+! (i)
$central long a$	longitude		-180 - +180	LONGITUDE	!!central long a!!
$central long b$	longitude		-180 - +180	!LONGITUDE	!!central long b!!
$compfail$	--	--	--	--	!!Compfail!!
$dest long$	longitude		E180-W180	LONGITUDE	!+dest long+! (i)
$dest mslp$	pressure	real	0-40.95 in. Hg	REAL	!+dest mslp pnl+! (i)
$Doppler coupled$	boolean	--	--	SIGN	!+Doppler coupled pnl+!
$drift angle filtered$	angle	integer	-180 - +180 (represents degrees)	SINT	!!drift angle filtered!!
$e coarse bias$	accel	real	-.02 - +.02 ft/sec/sec	SFRAC	!+E coarse bias+!
$e coarse scale$	speed	real	0, or .026 - .038 ft/sec	UFRAC	!!E coarse scale!!
$e fine bias$	accel	real	-.01 - +.01 ft/sec/sec	SFRAC	!+E fine bias+!
$e fine scale$	speed	real	0, or .00026-.00038 ft/sec	UFRAC	!!E fine scale!!
$elapsed navaln time$	timeint		0 - #navaln wrap-around#	TIME	!+elapsed navaln time+!
$fpangl at rls$	angle	real	-180 - +180 deg	REAL	!!fpangl at rls!!
$gndspd filtered$	speed	integer	0 - 1214 (represents knots)	UINT	!!gndspd filtered!!
$gyro drift delta n$	real		-.99 - +.99	SFRAC	!+gyro drift delta n+!
$heading MAG$	angle		0 - 360	ANGLE	!+heading MAG+!
$IMS diags2$	various	char_string		CHARSTR	!!IMS diags2!!
$L-probe$	boolean	--	--	SIGN	!+L-probe+!
$land based$	boolean	--	--	SIGN	!+Land based pnl+!
$longitude error$	longitude		E180-W180	LONGITUDE	!+longitude error+!
$longitude$	longitude		E180-W180	LONGITUDE	!+longitude+!
$low lat ct a$	integer		-90 - +90	SINT	!!low lat ct a!!
$low lat ct b$	integer		-90 - +90	SINT	!!low lat ct b!!
$mag variation$	angle		E180-W180	LONGITUDE	!+Mag variation pnl+!(i)
$map longitude$	longitude		E180-W180	LONGITUDE	!+Map longitude+!

TABLE 86. Display in Lower Panel Window (Sheet 2 of 3)

!+pnl config+!	Input Type	Output type (if different)	Value Constraints	!+low win fmt+!	!+low win val+!
$map orient a$	angle	boolean -- See note 2		SIGN	!!map orient a!!
$map orient b$	angle	boolean -- See note 2		SIGN	!!map orient b!!
$map sw diags$	various	char_string		CHARSTR	!!Map sw diags!!
$mark long$	longitude		E180-W180	LONGITUDE	!+mark long+! (i)
$n coarse bias$	accel	real	-.02 - +.02 ft/sec/sec	SFRAC	!+N coarse bias+!
$n coarse scale$	speed	real	0, or .026 - .038 ft/sec	UFRAC	!!N coarse scale!!
$n fine bias$	accel	real	-.01 - +.01 ft/sec/sec	SFRAC	!+N fine bias+!
$n fine scale$	speed	real	0, or .00026-.00038 ft/sec	UFRAC	!!N fine scale!!
$nav diags2$	various	char_string		CHARSTR	!!Nav diags2!!
$none$	--	--	--	--	!!blanks!!
$norm accel at rls$	accel	real	0 - 32767 g	REAL	!!norm accel at rls!!
$offset brg$	angle		0-360	ANGLE	!+offset brg pnl+! (i)
$OFP ver2$	char_string	--	--	CHARSTR	!!OFP version lower!!
$priority alt display$	various	char, integer		CHAR_UINT	!!priority alt display!!
$radalt priority$	boolean	--	--	SIGN	!+Radalt priority pnl+!
$SINS dhdg$	angle		0-360	ANGLE	!+SINS dhdg pnl+!
$SINS east vel$	speed	integer	-2047 - +2047kts	SINT	!+SINS east vel+!
$SINS heading$	angle		0-360	ANGLE	!+SINS heading+!
$SINS long$	longitude		E180-W180	LONGITUDE	!+SINS long+!
$SINS north vel$	speed	integer	-2047 - +2047kts	SINT	!+SINS north vel+!
$SINS valid2$	various	char_string		CHARSTR	!!SINS valid2!!
$SINS y offset$	distance	integer	-2047 - +2047 ft	SINT	!+SINS y offset pnl+!
$TAS ADC$	speed	integer	0 - 1214 knots	UINT	!!TAS ADC!!
$time to dest$	timeint		0:0:0-6:45:00; see note 1	TIME	!!time to dest!!
$v coarse bias$	accel	real	-.02 - +.02 ft/sec/sec	SFRAC	!+V coarse bias+!
$v coarse scale$	speed	real	0, or .026 - .038 ft/sec	UFRAC	!!V coarse scale!!
$WEAPTYP$	integer		0 - 99	UINT	!+WEAPTYP+!
$wind dir$	angle	integer	0 - 360	ANGLE	!!wind dir!!
$wpn sw diags$	various	char_string		CHARSTR	!!Wpn sw diags!!

TABLE 86. Display in Lower Panel Window (Sheet 3 of 3)

!+pnl config+!	Input Type	Output type (if different)	Value Constraints	!+low win fmt+!	!+low win val+!
$x corr increm$	angle	real	.32 - .48 deg	UFRAC	!+X corr increm+!
$x drift$	angrate	real	-1 - +1 deg/hour	SFRAC	!+X drift+!
$y corr increm$	angle	real	.32 - .48 deg	UFRAC	!+Y corr increm+!
$y drift$	angrate	real	-1 - +1 deg/hour	SFRAC	!+Y drift+!
$z corr increm$	angle	real	.32 - .48 deg	UFRAC	!+Z corr increm+!
$z drift$	angrate	real	-1 - +1 deg/hour	SFRAC	!+Z drift+!

Notes:

1. If !!time to dest!! is greater than 6:45:00, then 0:00:00 is displayed.
2. The mapping between types "angle" and "boolean" is as follows: The boolean value is true iff the angular measure is 0
3. For all values indexed with *(i)* (e.g., !+offset brg pnl+! (i)), there are several different versions of each that may be displayed. The version that should be displayed is denoted in the specifications by "i", where "i" stands for !+dest entry pnl+!, #multval lbound# <= i <= #multval hbound#.

10.3.5 Local Dictionary

Any term of the form !!item a!! The value of !+item+!, parameterized by the map value A. For instance, !!map orient a!! is defined as !+map orient+! parameterized by the map value A.

Any term of the form !!item b!! The value of !+item+! parameterized by the map value B. For instance, !!low lat ct b!! is defined as !+low lat ct+! parameterized by the map value B.

!!align stage!! If the current !+align stage+! is FM or HS or $None$, this display is blank. If it is TS then FG is displayed. Otherwise, the alignment stage is displayed in positions three and four of an otherwise-blank six-character string. (The third character of the name, if any, is truncated.)

!!Alt AGL at rls!! An altitude measure taken at the time of the first weapon release in the most recent stik. The source of the measure is !+alt priority ranging+!. The value is updated whenever the event @T(!+rls pts passed+! = 1) occurs.

!!altitude baro above tgt!! Defined by Table 87. In the table, !+dest altitude pnl+! and !+offset dht pnl+! are indexed by !+Fly to num+!.

TABLE 87. Value of !!altitude baro above tgt!!

MODES	!!altitude baro above tgt!!
BOCoffset *HUDdown2* *Noffset* *SBOCoffset* *SHUDdown2* *Snoffset*	!+alt ADC+! - !+dest altitude pnl+! - !+offset dht pnl+!
Any other weapon mode	!+alt ADC+! - !+dest altitude pnl+!

!!alt baro AGL!! Initialized to zero. Subsequent values defined in Table 88.

TABLE 88. Value of !!alt baro AGL!!

MODES	EVENTS	
AflyUpd	X	@T(!+new dest entry pnl entered+!) OR @T(!+TD pressed+!)
BOCFlyto0 *HUDdown1* *HUDdown2* *Nattack* *Noffset* *SBOCFlyto0* *SHUDdown1* *SHUDdown2* *Snattack* *Snoffset*	@F(!+desig+!) OR @T(In mode)	@T(!+desig+!) OR (@F(!+Slew displacement non-zero+!) WHEN(!+rls pts passed+! = 0))
BOC *SBOC*	@T(In mode AND !+gr ac tgt+! > 30 nmi)	@T(!+gr ac tgt+! <= 30 nmi) OR @F(!+Slew displacement non-zero+!) WHEN(!+gr ac tgt+! <= 20 nmi AND !+rls pts passed+! = 0)
BOCoffset *SBOCoffset*	@T(In mode) OR @F(!+desig+!) OR @T(!+gr ac tgt+! > 30 nmi)	@T(!+desig+!) OR @T(!+gr ac oap+! <= 30 nmi) OR @F(!+Slew displacement non-zero+!) WHEN(!+gr ac oap+! <= 20 nmi AND !+rls pts passed+! = 0)
CCIP	@T(In mode)	@T(!+ip elev+! <= 16 deg) WHEN(!+rls pts passed+! = 0)
!!alt baro AGL!!	0 feet	!!altitude baro above tgt!!

!!az miss dist at rls!! The value of !+az miss dist+! at the first release of the most recent stik, or zero. The value and when it the value changes is defined by Table 89. Value is initialized to zero.

TABLE 89. Value of !!az miss dist at rls!!

MODES	EVENTS	
BOC *BOCFlyto0* *BOCoffset* *HUDdown1* *HUDdown2* *Nattack* *Noffset* *SBOC* *SBOCFlyto0* *SBOCoffset* *SHUDdown1* *SHUDdown2* *Snattack* *Snoffset*	X	@T(!+rls pts passed+! = 1)
A/A Manrip *CCIP* *Manrip*	@T(In mode)	X
!!az miss dist at rls!!	0 feet	!+az miss dist+!

!!blanks!! Display all blanks in window.

!!Compfail!! The display for this item consists of a blank panel, with all the format lights turned on. There is no input item associated with this display.

!!drftangl IMS!! 0 if !+in ims_fail+!; !+drift angle IMS+! otherwise.

!!drift angle filtered!! !+sm drftang DRS+! if !+Doppler up+!; otherwise 0.

!!E coarse scale!! !+E coarse scale+! if !+E coarse scale+! != #IMSR init coarse vscale#; 0 feet/sec/pulse otherwise.

!!E fine scale!! !+E fine scale+! if !+E fine scale+! != #IMSR init fine vscale#; 0 feet/sec/pulse otherwise.

!!fpangl at rls!! The value displayed is !+flight path angle+!. The value is changed when @T(!+rls pts passed+! = 1) occurs when in one of the following weapon modes: *A/A Manrip*, *BOC*, *BOCFlyto0*, *BOCoffset*, *CCIP*, *HUDdown1*, *HUDdown2*, *Manrip*, *Nattack*, *Noffset*, *SBOC*, *SBOCFlyto0*, *SBOCoffset*, *SHUDdown1*, *SHUDdown2*, *Snattack*, or *Snoffset*.

!!gndspd filtered!! !+sm gndspd DRS+! if !+Doppler up+!; otherwise 0 fps.

!!groundspeed IMS!! 0 fps if !+in ims_fail+!; !+IMS horiz velocity+! otherwise.

!!IMS diags1!! This is a six-element string of character literals. The conditions for each element are given below. If a condition (or boolean input item) is true, then the element has the value 1; if false, 0.

TABLE 90. !!IMS diags1!!

Element #	Condition
1	!+IMS rel+!
2	!+WOG+!
3	!+Master Arm+!
4	!+IMS mode+! = $Gndal$
5	!+IMS mode+! = $Norm$
6	!+IMS mode+! = $Iner$

!!IMS diags2!! This is a seven-element string of character literals. The conditions for each element are given below. If a condition (or boolean input item) is true, then the element is set to 1; if false it is set to 0.

TABLE 91. !!IMS diags2!!

Element #	Condition
1	!+IMS mode+! = $Grid$
2	!+IMS mode+! = $Magsl$
3	NOT !+IMS ready+!
4	!+Auto-cal sw+!
5	!+ADCFAIL+!
6	!+Weap Pairs+!
7	Always set to $blank$

!!IMS total vel!! 0 fps if !+in ims_fail+!; otherwise, !+IMS total velocity+!.

!!Map sw diags!! This is a seven-element string of character literals. The conditions for each element are given below. If a condition (or boolean input item) is true, then the element is set to 1; otherwise it is set to 0.

TABLE 92. !!Map sw diags!!

Element #	Condition
1	!+Map scale sw+! = 80
2	!+Map hold+!
3	!+Map decenter+!
4	!+Map north-up+!
5	!+Map ldg+!
6	!+HUD reliable+!
7	Always set to $blank$

!!MFSW diags!! This is a six-element string of character literals. The conditions for each element are given below. If a condition is true, then the element is set to 1; otherwise it is set to 0.

TABLE 93. !!MFSW diags!!

Element #	Condition
1	!+High Drag+!
2	!+Weapon Mode+! = $NATT$ OR $NATTOFF$
3	!+Weapon Mode+! = $NATTOFF$ OR $BOCOFF$
4	!+Weapon Mode+! = BOC OR $BOCOFF$
5	!+Weapon Mode+! = $CCIP$
6	!+Weapon Mode+! = TF

!!N coarse scale!! !+N coarse scale+! if value != #IMSR init coarse vscale#; 0 feet/sec/pulse otherwise.

!!N fine scale!! !+N fine scale+! if value != #IMSR init fine vscale#; 0 feet/sec/pulse otherwise.

!!Nav diags1!! This is a six-element string of character literals. The conditions for each element are given below, followed by the element value corresponding to a true condition, and the element value corresponding a false condition..

TABLE 94. !!Nav diags1!!

Element #	Condition	Value if true	Value if false
1	NOT !+WOG+!	1	0
2	Always	$blank$	
3	(!+in mag_sl+! OR !+in grid+! OR !+in ims_fail+!) OR (!+in airaln+! AND !+align stage+! = FM)	1	0
4	Always	$blank$	
5	NOT !+IMS ready+!	1	0
6	Always	$blank$	

!!Nav diags2!! This is a seven-element string of character literals. The conditions for each element are given below, followed by the element value corresponding to a true condition, and the element value corresponding a false condition..

TABLE 95. !!Nav diags2!!

Element #	Condition	Value if true	Value if false
1	NOT !+IMS up+!	1	0
2	Always	$blank$	
3	NOT !+IMS reasonable+!	1	0
4	Always	$blank$	
5	NOT !+drift angle reliable+! OR NOT !+gnd speed reliable+!	1	0
6	Always	$blank$	
7	NOT !+Doppler reasonable+!	1.	0

!!norm accel at rls!! The value displayed is !+normal accel+!. The value is changed when @T(!+rls pts passed+! = 1) occurs when in one of the following weapon modes: *A/A Manrip*, *BOC*, *BOCflyto0*, *BOCoffset*, *CCIP*, *HUDdown1*, *HUDdown2*, *Manrip*, *Nattack*, *Noffset*, *SBOC*, *SBOCflyto0*, *SBOCoffset*, *SHUDdown1*, *SHUDdown2*, *Snattack*, or *Snoffset*.

!!North light!! To achieve this display, set !+N Light+! = $true$ and set !+Format U321+! = $true$.

!!OFP version upper!! Two character strings to be displayed in the panel display windows denoting information about this OFP. Defined at system generation time by the sysgen parameters #OFP version upper# and #OFP version lower#, respectively.

!!OFP version lower!! Two character strings to be displayed in the panel display windows denoting information about this OFP. Defined at system generation time by the sysgen parameters #OFP version upper# and #OFP version lower#, respectively.

!!priority alt display!! The value of !+alt priority stale+!, prefixed by the character form of !+alt priority source+!.

!!resolution!! The required display resolution for each display item is given in the following table.

TABLE 96. !!Resolution!! (Sheet 1 of 3)

Display item:	Display !!resolution!!:
!!Alt AGL at rls!!	1 foot
!!az miss dist at rls!!	1 foot
!+az ref hdg pnl+!	1 minute
!!alt baro AGL!!	1 foot
!+burst ht pnl+!	2 feet (rounded down value)
!+central long a pnl+!	1 second
!+central long b pnl+!	1 second
!+data nbr pnl+!	1 (integer)
!+dest altitude pnl+!	2 feet (rounded down value)
!+dest lat+!	1 second
!+dest long+!	1 second
!+dest mslp pnl+!	.01 inches
!+drift angle IMS+!	1 degree
!!drift angle filtered!!	1 degree
!+E coarse bias+!	.0003 feet/sec/sec
!!E coarse scale!!	.00003 feet/sec/pulse
!+E fine bias+!	.0003 feet/sec/sec
!!E fine scale!!	.000001 feet/sec/pulse
!+E vel IMS+!	1 knot
!+elapsed navaln time+!	1 second
!!fpangl at rls!!	.01 degrees
!!gndspd filtered!!	1 knot
!!groundspeed IMS!!	1 knot
!+gyro drift delta n+!	.001 deg/hour
!+hdg system+!	1 minute

TABLE 96. !!Resolution!! (Sheet 2 of 3)

Display item:	Display !!resolution!!:
!+heading IMS+!	1 minute
!+heading MAG+!	1 minute
!!IMS total vel!!	1 knot
!+latitude+!	1 second
!+latitude error+!	1 second
!+longitude+!	1 second
!+longitude error+!	1 second
!+low lat ct a pnl+!	10 seconds
!+low lat ct b pnl+!	10 seconds
!+Mag variation pnl+!	1 second
!+Map latitude+!	1 second
!+Map longitude+!	1 second
!+mark lat+!	1 second
!+mark long+!	1 second
!+N coarse bias+!	.0003 feet/sec/sec
!!N coarse scale!!	.00003 feet/sec/pulse
!+N fine bias+!	.0003 feet/sec/sec
!!N fine scale!!	.000001 feet/sec/pulse
!+N vel IMS+!	1 knot
!!norm accel at rls!!	.1 g
!+offset brg pnl+!	1 second
!+offset dht pnl+!	2 feet (rounded down value)
!+offset rng pnl+!	4 feet (rounded down value)
!+SINS dhdg pnl+!	1 second
!+SINS east vel+!	1 knot
!+SINS heading+!	1 minute
!+SINS lat+!	1 second
!+SINS long+!	1 second
!+SINS north vel+!	1 knot
!+SINS x offset pnl+!	1 foot
!+SINS y offset pnl+!	1 foot
!+SINS z offset pnl+!	1 foot
!!Slant range at rls!!	8 feet
!!TAS ADC!!	1 knot
!!TAS ADC at rls!!	1 knot
!!time to dest!!	1 second

TABLE 96. !!Resolution!! (Sheet 3 of 3)

Display item:	Display !!resolution!!:
!+V coarse bias+!	.0003 feet/sec/sec
!!V coarse scale!!	.00003 feet/sec/pulse
!!vel E!!	1 knot
!!vel N!!	1 knot
!!Weap Interval!!	10 feet
!!Weap Quantity!!	1
!+WEAPTYP+!	1
!!wind dir!!	1 second
!!wind speed!!	1 knot

!!SINS valid1!! This is a six-element string of character literals. The conditions for each element and the corresponding display characters are given below. Elements not specified in the table are always blank.

TABLE 97. !!SINS valid1!!

Element #	Condition	Value if true	Value if false
1	!+SINS heading valid+!	0	1
2	!+SINS north vel valid+!	0	1
3	!+SINS roll valid+!	0	1

!!SINS valid2!! This is a seven-element string of character literals. The conditions for each element and the corresponding display characters are given below. Elements not specified in the table are always blank.

TABLE 98. !!SINS valid2!!

Element #	Condition	Value if true	Value if false
1	!+SINS east vel valid+!	0	1
3	!+SINS pitch valid+!	0	1
5	!+SINS lat valid+!	0	1
7	!+SINS long valid+!	0	1

!!Slant range at rls!! Initialized to zero; subsequent values are defined by table below.

TABLE 99. !!Slant range at rls!!

MODES	EVENTS		
BOC *BOCFlyto0* *BOCoffset* *HUDdown1* *HUDdown2* *Nattack* *Noffset* *SBOC* *SBOCflyto0* *SBOCoffset* *SHUDdown1* *SHUDdown2* *Snattack* *Snoffset*	X	@T(!+rls pts passed+! = 1)	X
CCIP	X	X	@T(!+rls pts passed+! = 1)
A/A Manrip *Manrip*	@T(In mode)	X	X
!!Slant range at rls!!	0 feet	!+sr ac tgt+!	!+sr ac ip+!

!!STARDY diags!! This is a six-element string of character literals. The conditions for each element are given below. If the boolean input item is true, then the element has the value 1; otherwise, 0.

TABLE 100. !!STARDY diags!!

Element #	Condition
1	!+Ready Stations+! element #1
2	!+Ready Stations+! element #2
3	!+Ready Stations+! element #3
4	!+Ready Stations+! element #4
5	!+Ready Stations+! element #5
6	!+Ready Stations+! element #6

!!TAS ADC!! !+TAS ADC+! if (!+in flight+! AND !+adc tas up+!); otherwise 0 fps.

!!TAS ADC at rls!! The value displayed is !+TAS ADC+!. The value is changed whenever @T(!+rls pts passed+! = 1) occurs when in one of the following weapon modes: *A/A Manrip*, *BOC*, *BOCflyto0*, *BOCoffset*, *CCIP*, *HUDdown1*, *HUDdown2*, *Manrip*, *Nattack*, *Noffset*, *SBOC*, *SBOCflyto0*, *SBOCoffset*, *SHUDdown1*, *SHUDdown2*, *Snattack*, or *Snoffset*.

!!time to dest!! Use !+time to ftpt+!.

!!V coarse scale!! !+V coarse scale+! if value is != #IMSR init coarse vscale#; 0 feet/sec/pulse otherwise.

!!vel E!! Initialized to zero. Subsequent values are defined by the table below. In *SINSaln* and *Sautocal* when @T(!+new align stage+!) occurs, the value is the SINS velocities for the first display, after which the display reverts to the IMS velocities.

!!vel N!! Initialized to zero. Subsequent values are defined by the table below. In *SINSaln* and *Sautocal* when @T(!+new align stage+!) occurs, the value is the SINS velocities for the first display, after which the display reverts to the IMS velocities.

TABLE 101. Values of !!vel E!! and !!vel E!!

MODES	EVENTS		
SINSaln *Sautocal*	@T(!+new align stage+!)	See note above	@F(In mode)
Landaln *Lautocal*	X	@T(In mode)	X
IMS fail	X	X	@T(In mode)
All other navigation or align. modes	X	@T(In mode)	X
!!vel E!!	!+SINS east vel+!	!+E vel IMS+!	0 fps
!!vel N!!	!+SINS north vel+!	!+N vel IMS+!	0 fps

!!Weap Interval!! The value of !+Weap Interval+! whenever !+rls pts passed+!=0. If the value of !+Weap Interval+! changes while !+rls pts passed+! != 0, !!Weap Interval!! does not change.

!!Weap Quantity!! The value of !+Weap Quantity+! whenever !+rls pts passed+!=0. If the value of !+DI.Weap Quantity+! changes while !+rls pts passed+! != 0, !!Weap Quantity!! does not change.

!!wind dir!! The direction of !+SS.wind vel+! in the North-East plane, measured clockwise (looking down) from North; 0 <= !!wind dir!! < 360.

!!wind speed!! The magnitude of !+wind vel+!.

!!Wpn sw diags!! This is a seven-element string of character literals. The conditions for each element and corresponding displays are given below. Elements not specified in the table are always blank.

TABLE 102. !!Wpn sw diags!!

Element #	Condition	Value if true	Value if false
1	!+Gun Enable+!	1	0
2	!+RE pressed+!	1	0
3	!+Mult Rack+!	1	0
4	!+TD pressed+!	1	0

11.0 Projected Map Display Set Functions

11.1 Set the map indicator

!+Map indicator+! = !+heading MAG+! + !+grtk+! - !+heading IMS+!

11.2 Set the map orientation angle

TABLE 103. Set the map orientation angle.

MODES	CONDITIONS		
All modes except *Grtest*	!+Map north-up+! AND NOT !+Map hold+!	NOT !+Map north-up+! AND NOT !+Map hold+!	!+Map hold+!
!+Map rotation+!	0	!+grtk+!	!!stale orient!!

11.2.1 Local Dictionary

!!stale orient!! Last value of map orientation angle before @T(!+Map hold+!).

11.3 Set the map pointer

TABLE 104. Set the map pointer.

MODES	CONDITIONS	
All modes	!+Fly to num+! = 0	!+Fly to num+! != 0
!+Map pointer angle+!	0	!+brg grtk ftpt+!

11.4 Set the map reference point

TABLE 105. Set the map reference point.

MODES	EVENTS	
All modes except *Grtest*	@T(!+Init complete+!) WHEN(!+Map north-up+! OR NOT !+Map decenter+!) OR @F(NOT !+Map north-up+! AND !+Map decenter+!)	@T(!+Init complete+!) WHEN(NOT !+Map north-up+! AND !+Map decenter+!) OR @T(NOT !+Map north-up+! AND !+Map decenter+!)
!+Map ref pt+!	$center$	$bottom-center$

11.5 Position the map

TABLE 106. Position the map

MODES	CONDITIONS	
All navigation and alignment modes; *MapUpd*	!!Posn displayable!!	NOT !!Posn displayable!!
!+Map latitutude+!	!!refpt lat!!	X
!+Map longitude+!	!!refpt long!!	X
!+Map position valid+!	$true$	$false$
!+Map warning+!	$false$	$true$

11.5.1 Local Dictionary

!!refpt!!, !!refpt lat!!, !!refpt long!! All defined by Table 107.

TABLE 107. Values of !!refpt!!, !!refpt lat!!, !!refpt long!!

MODES	CONDITIONS		
Not *MapUpd*	NOT !!Dest/Mark!! AND NOT !+Map hold+! AND NOT !+during slewing+!	NOT !+Map hold+! AND NOT !+during slewing+! AND !!Dest/Mark!!	!+Map hold+! OR !+during slewing+!
MapUpd	NOT !+desig+! AND NOT !+during slewing+!	X	!+desig+! OR !+during slewing+!
!!refpt!!	a/c present posn	!!Recalled pt!!	!!Slewed-to point!!
!!refpt lat!!	!+latitude+!	!!Recalled lat!!	!!slewed map lat!!
!!refpt long!!	!+longitude+!	!!Recalled long!!	!!slewed map long!!

!!Dest/Mark!! !!Dest displayed!! OR !!Mark displayed!!.

!!Dest displayed!! True between the occurrence of either @T(!+pnl config+! = $dest lat$) or @T(!+pnl config+! = $dest long$) and the next occurrence of @T(!+pnl config changed+!). False at all other times.

!!Mark displayed!! True between the occurrence of either @T(!+pnl config+! = $mark lat$) or @T(!+pnl config+! = $mark long$) and the next occurrence of @T(!+pnl config changed+!). False at all other times.

!!Posn displayable!! !+Map displayable+!.

!!Recalled pt!!, !!Recalled lat!!, !!Recalled long!! Defined by Table 108. If NOT !!Dest displayed!! AND NOT !!Mark displayed!! then !!Recalled pt!!, !!Recalled lat!!, and !!Recalled long!! have no meaning. In the table, !+mark lat+! and !+mark long+! are understood to be indexed by !+dest entry pnl+!.

TABLE 108. Values of !!Recalled pt!!, !!Recalled lat!!, !!Recalled long!!

MODES	CONDITIONS	
All modes	!!Dest displayed!!	!!Mark displayed!!
!!Recalled pt!!	called-up point	mark location
!!Recalled lat!!	!+latitude cup+!	!+mark lat+!
!!Recalled long!!	!+longitude cup+!	!+mark long+!

!!Slewed-to point!! The position defined by !!slewed map lat!! and !!slew map long!!, respectively. This is the point that the map has moved to because of inputs from the slew control.

!!slewed map lat!! The latitude of the new map position, computed by adding !+slew map delta lat+! to the previous latitude of the map display.

!!slewed map long!! The longitude of the new map position, computed by adding !+slew map delta long+! to the previous longitude of the map display.

11.6 Set the map scale

TABLE 109. Set the map scale.

MODES	EVENTS	
All modes	@T(!+Init complete+!) WHEN(NOT !+Map scale sw+!) OR @F(!+Map scale sw+!)	@T(!+Init complete+!) WHEN(!+Map scale sw+!) OR @T(!+Map scale sw+!)
!+Map scale+!	#Map scale array#(1)	#Map scale array#(2)

11.7 Set the map reconfiguration values

TABLE 110. !+central long a+

MODES	EVENTS
All modes except *Grtest*	@T(!+new central long a pnl entered+!)
!+central long a+!	!+central long a pnl+!

TABLE 111. !+central long b+!

MODES	EVENTS
All modes except *Grtest*	@T(!+new central long b pnl entered+!)
!+central long b+!	!+central long b pnl+!

TABLE 112. !+low lat ct a+!

MODES	EVENTS
All modes except *Grtest*	@T(!+new low lat ct a pnl entered+!)
!+low lat ct a+!	!+low lat ct a pnl+!

TABLE 113. !+low lat ct b+!

MODES	EVENTS
All modes except *Grtest*	@T(!+new low lat ct b pnl entered+!)
!+low lat ct b+!	!+low lat ct b pnl+!

TABLE 114. !+map orient a+!

MODES	EVENTS
All modes except *Grtest*	@T(!+new map orient a pnl entered+!)
!+map orient a+!	!+map orient a pnl+!

TABLE 115. !+map orient b+!

MODES	EVENTS
All modes except *Grtest*	@T(!+new map orient b pnl entered+!)
!+map orient b+!	!+map orient b pnl+!

12.0 Shipboard Inertial Navigations System Functions

12.1 Start and stop the SINS

TABLE 116. Start and stop the SINS

MODES	EVENTS	
All alignment and navigation modes	@T(!+in flight+!) OR @T(!+Land based pnl+!)	@T(!+Init complete+!) WHEN (NOT !+Land based pnl+! AND NOT !+in flight+!) OR @F(!+Land based pnl+!) WHEN (NOT !+in flight+!) OR @F(!+in flight+!) WHEN (NOT !+Land based pnl+!)
!+SINS enabled+!	$false$	$true$

13.0 Visual Indicator Functions

13.1 Control the Auto-cal indicator

TABLE 117. Control the Auto-cal indicator

MODES	EVENTS	
Lautocal *Sautocal*	@T(In mode)	@F(In mode)
All other alignment or navigation modes	X	@T(In mode)
!+Auto-cal+!	On	Off

13.2 Control the *IMS Non-aligned* indicator

TABLE 118. Control the *IMS Non-aligned* indicator.

MODES	EVENTS			
Lautocal *Sautocal*	@T(In mode)	@F(In mode)	X	X
Landaln	@T(In mode)	@T(!+land velocity test passed+!)	X	X
SINSaln	@T(In mode)	@T(!+SINS velocity test passed+!) OR @F(In mode)	@T(In mode AND NOT !!SINS up!!)	@T(!!SINS up!!)
Airaln	@F(!+align stage = CL) OR @T(In mode) WHEN(!+CL stage complete+!)	@T(In mode) WHEN (NOT !+CL stage complete+!) OR @T(!+air velocity test passed+!)	@T(In mode AND NOT !!All IMS cks passed!!)	@T(!!All IMS cks passed!!)
01 Update	@T(!+land velocity test failed+!) OR @T(!+drift test failed+!)	@T(!+land velocity test passed+! AND !+drift test passed+!)	X	X
HUDaln	@F(!+align stage+! = HS) WHEN (!+IMS mode+! = $Gndal$)	@T(In mode) OR @F(In mode) OR @F(!+align stage+! = CL)	X	X
DI *DIG* *PolarDI*	@T(!+nav velocity test failed+!)	X	@T(In mode AND NOT !!All IMS cks passed!!)	@T(!!All IMS cks passed!!)
Grid *IMS fail* *Mag sl*	X	@T(In mode)	X	X
I *PolarI* *UDI*	X	X	@T(In mode AND NOT !!All IMS cks passed!!)	@T(!!All IMS cks passed!!)
!+Non-align+!	On	Off	$Intermittent$	!!stale mode!!

Note: $Intermittent$ takes precedence. Upon !!stale mode!!, the light goes to the Output value for which the condition has most recently been met. (The state may have changed several times while the Output value was $Intermittent$.)

13.2.1 Local Dictionary

!!All IMS cks passed!! (!!IMS-Dop Reasonable!! OR NOT !+Doppler up+!) AND (!!IMS-ADC Reasonable!! OR NOT !+adc tas up+!) AND !+IMS reasonable+!

!!stale mode!! The value of !+Non-align+! before the last occurrence of @T(!+Non-align+! = $Intermittent$)

!!IMS-ADC Reasonable!! ABS(!+IMS total velocity+! - !+TAS ADC+!) <= 191 knots.

!!IMS-Dop Reasonable!! ABS(!+IMS horiz velocity+! - !+gnd speed DRS+!) <= 62 knots.

!!SINS up!! At least one validity boolean must be true AND the SINS velocity cannot be invalid for more than one second AND the SINS attitude data cannot be invalid for more than three seconds. If any of these conditions are violated, then the value is false.

13.3 Set the visual indicator blink periods

This function is never performed, because under the current requirements, the visual indicator blink periods never need to be changed from their default values.

14.0 Weapon Release System Functions

14.1 Release a weapon

TABLE 119. Prepare a Weapon

MODES	EVENTS
All weapon modes	@T(!+time to prepare+!) WHEN(!+RE pressed+!)
!+prepare weapon+!	!! signal prepare weapon!!

TABLE 120. Release a Weapon

MODES	EVENTS
NBnotShrike **NBShrike** **HiNuke** **LoNuke**	@T(!+computed rls+!) WHEN(!+RE pressed+!)
A/A Manrip *CCIP* *Manrip* *Walleye*	@T(!!RE delay!!) WHEN(!+wpns rlsd+! = 0) OR @T(!+computed rls+!) WHEN(!+RE pressed+!)
!+release weapon+!	!!signal release weapon!!
!+set release pulse width+!	!+release pulse width+!

14.1.1 Local Dictionary

!!RE delay!! If !+preparation time+! is defined for the current weapon type, then this term is true iff the elapsed time since the last occurrence of @T(!+RE pressed+!) >= the value of !+preparation time+! obtained when @T(!+RE pressed+!) occurred.

!!signal prepare weapon!! NOT !+prepare weapon+!

!!signal release weapon!! NOT !+release weapon+!

15.0 Ground Test Functions

15.1 Conduct the computer self-test

TABLE 121. Conduct the computer self-test

MODES	EVENTS	
Grtest	@T(!+power up+!)	@F(!+memory test result+!) WHEN (!+test stage+! = CS) OR @F(!+interrupt test result+!) WHEN (!+test stage+! = TM) OR @F(!+timer test result+!) WHEN (!+test stage+! = TM) OR @F(!+XACC test result+!) WHEN (!+test stage+! = GA) OR @F(!+YACC test result+!) WHEN (!+test stage+! = GA) OR @F(!+ZACC test result+!) WHEN (!+test stage+! = GA) OR @F(!+DIOW1 test result+!) WHEN (!+test stage+! = DIO) OR @F(!+DIOW2 test result+!) WHEN (!+test stage+! = DIO) OR @F(!+DIOW3 test result+!) WHEN (!+test stage+! = DIO) OR @F(!+CSA test result+!) WHEN (!+test stage+! = SC) OR @F(!+CSB test result+!) WHEN (!+test stage+! = SC) OR @F(!+DC test result+!) WHEN (!+test stage+! = DC) OR @F(!+AC test result+!) WHEN (!+test stage+! = AC)
!+failed state+!	FALSE	TRUE

16.0 Modes of Operation

16.1 Transition Between Alignment/Test/Navigation Modes

TABLE 122. Initial Navigation, Alignment, and Test Modes

Conditions	Modes
!+IMS up+! AND !+IMS mode+!=$Gndal$ AND !+land based pnl+! = $true$	*Landaln*
!+IMS up+! AND (!+IMS mode+!=$Iner$ OR $Norm$ OR (!+IMS mode+!=$Gndal$ AND !+land based pnl+! = $false$))	*OLB*
!+IMS up+! AND !+IMS mode+!=$Mag sl$	*Mag sl*
!+IMS up+! AND !+IMS mode+!=$Grid$	*Grid*
NOT !+IMS up+!	*IMS fail*

TABLE 123. Transitions between Alignment, Navigation, and Test Modes while the aircraft is not airborne (!+in flight+! = $false$)

Current Mode	!+new posn entered+!	!+in flight+!	!+Auto-cal sw +! = $true$	!+desig+!	!+land based panel+! = $false$	!+CA2 stage complete+!	!+CL2 stage complete+!	!+ND2 stage complete+!	!+HS stage complete+!	!+Self-test+! = $true$	!+IMS up+!	!+latitude+! > 80	!+IMS mode+! =$Gndal$	!+IMS mode+! =$Norm$	!+IMS mode+! =$Iner$	!+IMS mode+! =$Mag sl$	!+IMS mode+! =$Grid$	!+SINS enabled+!	!+SINS velocity test passed+!	!+land velocity test passed+!	!+No intervening takeoff+!	!+ground tests finished+!	New Mode
Lautocal	@T	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	*Landaln*
	-	-	-	-	-	-	-	@T	-	f	-	f	-	-	-	-	-	-	-	-	-	-	*I*
	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	*OLB*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	
	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	*Mag sl*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	*Grid*
	-	-	-	-	-	-	-	-	-	f	@F	-	-	-	-	-	-	-	-	-	-	-	*IMS fail*
	-	-	-	-	-	-	-	@T	-	f	-	t	-	-	-	-	-	-	-	-	-	-	*PolarI*
	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	*Grtest*
Sautocal	@T	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	*SINSaln*
	-	-	-	-	-	-	-	@T	-	f	-	f	-	-	-	-	-	-	-	-	-	-	*I*
	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	*OLB*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	
	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	f	-	-	-	-	-	-	f	-	f	-	-	-	-	-	-	-	@F	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	*Mag sl*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	*Grid*
	-	-	-	-	-	-	-	-	-	f	@F	-	-	-	-	-	-	-	-	-	-	-	*IMS fail*
	-	-	-	-	-	-	-	@T	-	f	-	t	-	-	-	-	-	-	-	-	-	-	*PolarI*
	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	*Grtest*

TABLE 124. Transitions between Alignment, Navigation, and Test Modes while the aircraft is not airborne (!+in flight+! = $false$)

Current Mode	!+new posn entered+!	!+in flight+!	!+Auto-cal sw +! = $true$	!+desig+!	!+land based panel+! = $false$	!+CA2 stage complete+!	!+CL2 stage complete+!	!+ND2 stage complete+!	!+HS stage complete+!	!+Self-test+! = $true$	!+IMS up+!	!+latitude+! > 80	!+IMS mode+! =$Gndal$	!+IMS mode+! =$Norm$	!+IMS mode+! =$Iner$	!+IMS mode+! =$Mag sl$	!+IMS mode+! =$Grid$	!+SINS enabled+!	!+SINS velocity test passed+!	!+land velocity test passed+!	!+No intervening takeoff+!	!+ground tests finished+!	New Mode
Landaln	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	*Lautocal*
	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*Landaln*
	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*HUDaln*
	-	@T	-	-	-	-	t	-	-	-	-	f	-	-	-	-	-	-	-	-	-	-	*I*
	-	-	-	-	@T	-	t	-	-	-	-	f	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	t	-	-	-	-	f	-	-	@T	-	-	-	-	-	-	-	
	-	-	-	-	-	-	t	-	-	-	-	f	-	@T	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	f	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	*OLB*
	-	-	-	-	-	-	f	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	
	-	-	-	-	@T	-	f	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	@T	-	-	-	-	f	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	*Mag sl
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	*Grid*
	-	-	-	-	-	-	-	-	-	f	@F	-	-	-	-	-	-	-	-	-	-	-	*IMS fail
	-	@T	-	-	-	-	t	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	*PolarI*
	-	-	-	-	@T	-	t	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	t	-	-	-	-	t	-	-	@T	-	-	-	-	-	-	-	
	-	-	-	-	-	-	t	-	-	-	-	t	-	@T	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	*Grtest*

TABLE 125. Transitions between Alignment, Navigation, and Test Modes while the aircraft is not airborne (!+in flight+! = $false$)

Current Mode	!+new posn entered+!	!+in flight+!	!+Auto-cal sw +! = $true$	!+desig+!	!+land based panel+! = $false$	!+CA2 stage complete+!	!+CL2 stage complete+!	!+ND2 stage complete+!	!+HS stage complete+!	!+Self-test+! = $true$	!+IMS up+!	!+latitude+! > 80	!+IMS mode+! =$Gndal$	!+IMS mode+! =$Norm$	!+IMS mode+! =$Iner$	!+IMS mode+! =$Mag sl$	!+IMS mode+! =$Grid$	!+SINS enabled+!	!+SINS velocity test passed+!	!+land velocity test passed+!	!+No intervening takeoff+!	!+ground tests finished+!	New Mode
SINSaln	-	-	@T	-	t	-	-	-	-	f	-	-	-	-	-	-	-	-	t	-	-	-	*Sautocal*
	-	-	-	-	@F	-	-	-	-	f	-	-	-	-	-	-	-	-	-	-	-	-	*Landaln*
	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*SINSaln*
	-	@T	-	-	-	-	t	-	-	-	-	f	-	-	-	-	-	-	-	-	-	-	*I*
	-	-	-	-	t	-	t	-	-	f	-	f	-	-	-	f	f	@F	-	-	-	-	
	-	-	-	-	-	-	t	-	-	-	-	f	-	-	@T	-	-	-	-	-	-	-	
	-	-	-	-	-	-	t	-	-	-	-	f	-	@T	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	f	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	*OLB*
	-	-	-	-	-	-	f	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	
	-	-	-	-	t	-	f	-	-	f	-	-	-	-	-	f	f	@F	-	-	-	-	
	-	@T	-	-	-	-	f	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	*Mag_sl*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	*Grid*
	-	-	-	-	-	-	-	-	-	f	@F	-	-	-	-	-	-	-	-	-	-	-	*IMS_fail*
	-	@T	-	-	-	-	t	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	*PolarI*
	-	-	-	-	t	-	t	-	-	f	-	t	-	-	-	f	f	@F	-	-	-	-	
	-	-	-	-	-	-	t	-	-	-	-	t	-	-	@T	-	-	-	-	-	-	-	
	-	-	-	-	-	-	t	-	-	-	-	t	-	@T	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	*Grtest*

TABLE 126. Transitions between Alignment, Navigation, and Test Modes while the aircraft is not airborne (!+in flight+! = $false$)

Current Mode	!+new posn entered+!	!+in flight+!	!+Auto-cal sw +! = $true$	!+desig+!	!+land based panel+! = $false$	!+CA2 stage complete+!	!+CL2 stage complete+!	!+ND2 stage complete+!	!+HS stage complete+!	!+Self-test+! = $true$	!+IMS up+!	!+latitude+! > 80	!+IMS mode+! =$Gndal$	!+IMS mode+! =$Norm$	!+IMS mode+! =$Iner$	!+IMS mode+! =$Mag sl$	!+IMS mode+! =$Grid$	!+SINS enabled+!	!+SINS velocity test passed+!	!+land velocity test passed+!	!+No intervening takeoff+!	!+ground tests finished+!	New Mode
01Update	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*Lautocal*
	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*Landaln*
	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*HUDaln*
	-	@T	-	-	-	-	-	-	-	-	-	f	-	-	-	-	-	-	-	-	-	-	*I*
	-	-	-	-	-	-	-	-	-	-	-	f	-	-	@T	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	f	-	@T	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	*Mag sl*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	*Grid*
	-	-	-	-	-	-	-	-	-	f	@F	-	-	-	-	-	-	-	-	-	-	-	*IMS fail*
	-	@T	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	*PolarI*
	-	-	-	-	-	-	-	-	-	-	-	t	-	-	@T	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	t	-	@T	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	*Grtest*

TABLE 127. Transitions between Alignment, Navigation, and Test Modes while the aircraft is not airborne (!+in flight+! = $false$)

Current Mode	!+new posn entered+!	!+in flight+!	!+Auto-cal sw +! = $true$	!+desig+!	!+land based panel+! = $false$	!+CA2 stage complete+!	!+CL2 stage complete+!	!+ND2 stage complete+!	!+HS stage complete+!	!+Self-test+! = $true$	!+IMS up+!	!+latitude+! > 80	!+IMS mode+! =$Gndal$	!+IMS mode+! =$Norm$	!+IMS mode+! =$Iner$	!+IMS mode+! =$Mag sl$	!+IMS mode+! =$Grid$	!+SINS enabled+!	!+SINS velocity test passed+!	!+land velocity test passed+!	!+No intervening takeoff+!	!+ground tests finished+!	New Mode
HUDaln	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	*Lautocal*
	-	@T	-	-	-	-	t	-	-	-	-	f	-	-	-	-	-	-	-	-	-	-	*I*
	-	-	-	-	-	-	t	-	-	-	-	f	-	-	@T	-	-	-	-	-	-	-	
	-	-	-	-	-	-	t	-	-	-	-	f	-	@T	-	-	-	-	-	-	-	-	
	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	*OLB*
	-	@T	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	-	
	-	@T	-	-	-	-	f	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	@T	-	-	-	-	t	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	t	-	-	-	-	-	-	-	
	-	-	-	-	-	-	f	-	t	-	-	-	-	@T	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	f	-	t	-	-	-	-	-	@T	-	-	-	-	-	-	-	
	-	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	*Mag sl*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	*Grid*
	-	-	-	-	-	-	-	-	-	f	@F	-	-	-	-	-	-	-	-	-	-	-	*IMS fail*
	-	@T	-	-	-	-	t	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	*PolarI*
	-	-	-	-	-	-	t	-	-	-	-	t	-	@T	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	t	-	-	-	-	t	-	-	@T	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	*Grtest*

TABLE 128. Transitions between Alignment, Navigation, and Test Modes while the aircraft is not airborne (!+in flight+! = $false$)

Current Mode	!+new posn entered+!	!+in flight+!	!+Auto-cal sw +! = $true$	!+desig+!	!+land based panel+! = $false$	!+CA2 stage complete+!	!+CL2 stage complete+!	!+ND2 stage complete+!	!+HS stage complete+!	!+Self-test+! = $true$	!+IMS up+!	!+latitude+! > 80	!+IMS mode+! =$Gndal$	!+IMS mode+! =$Norm$	!+IMS mode+! =$Iner$	!+IMS mode+! =$Mag sl$	!+IMS mode+! =$Grid$	!+SINS enabled+!	!+SINS velocity test passed+!	!+land velocity test passed+!	!+No intervening takeoff+!	!+ground tests finished+!	New Mode
I	-	-	-	-	f	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	f	-	-	*Landaln*
	@T	-	-	-	f	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	
	@T	-	-	-	t	-	-	-	-	-	-	-	t	-	-	-	-	t	-	-	-	-	*SINSaln*
	-	-	-	-	f	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	t	f	-	*01Update*
	-	-	-	@T	f	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*HUDaln*
	@T	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	-	*OLB*
	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	
	@T	-	-	-	t	-	-	-	-	-	-	-	t	-	-	-	-	f	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	*Mag sl*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	*Grid*
	-	-	-	-	-	-	-	-	-	f	@F	-	-	-	-	-	-	-	-	-	-	-	*IMS_fail*
	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	*PolarI*
	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	*Grtest*
OLB	-	-	-	-	f	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	*Landaln*
	@T	-	-	-	t	-	-	-	-	-	-	-	t	-	-	-	-	t	-	-	-	-	*SINSaln*
	-	-	-	@T	f	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*HUDaln*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	*Mag_sl*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	*Grid*
	-	-	-	-	-	-	-	-	-	f	@F	-	-	-	-	-	-	-	-	-	-	-	*IMS_fail*
	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	*Grtest*

TABLE 129. Transitions between Alignment, Navigation, and Test Modes while the aircraft is not airborne (!+in flight+! = $false$)

Current Mode	!+new posn entered+!	!+in flight+!	!+Auto-cal sw +! = $true$	!+desig+!	!+land based panel+! = $false$	!+CA2 stage complete+!	!+CL2 stage complete+!	!+ND2 stage complete+!	!+HS stage complete+!	!+Self-test+! = $true$	!+IMS up+!	!+latitude+! > 80	!+IMS mode+! =$Gndal$	!+IMS mode+! =$Norm$	!+IMS mode+! =$Iner$	!+IMS mode+! =$Mag sl$	!+IMS mode+! =$Grid$	!+SINS enabled+!	!+SINS velocity test passed+!	!+land velocity test passed+!	!+No intervening takeoff+!	!+ground tests finished+!	New Mode
Mag_sl	-	-	-	-	f	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	*Landaln*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	*OLB*
	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	
	-	-	-	-	t	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	*Grid*
	-	-	-	-	-	-	-	-	-	f	@F	-	-	-	-	-	-	-	-	-	-	-	*IMS_fail*
	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	*Grtest*
Grid	-	-	-	-	f	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	*Landaln*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	*OLB*
	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	
	-	-	-	-	t	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	*Mag_sl*
	-	-	-	-	-	-	-	-	-	f	@F	-	-	-	-	-	-	-	-	-	-	-	*IMS_fail*
	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	*Grtest*
IMS_fail	-	-	-	-	f	-	-	-	-	-	@T	-	t	-	-	-	-	-	-	-	-	-	*Landaln*
	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	t	-	-	-	-	-	-	-	*OLB*
	-	-	-	-	-	-	-	-	-	-	@T	-	-	t	-	-	-	-	-	-	-	-	
	-	-	-	-	t	-	-	-	-	-	@T	-	t	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	t	-	-	-	-	-	-	*Mag_sl*
	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	t	-	-	-	-	-	*Grid*
	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	*Grtest*

TABLE 130. Transitions between Alignment, Navigation, and Test Modes while the aircraft is not airborne (!+in flight+! = $false$)

Current Mode	!+new posn entered+!	!+in flight+!	!+Auto-cal sw +! = $true$	!+desig+!	!+land based panel+! = $false$	!+CA2 stage complete+!	!+CL2 stage complete+!	!+ND2 stage complete+!	!+HS stage complete+!	!+Self-test+! = $true$	!+IMS up+!	!+latitude+! > 80	!+IMS mode+! =$Gndal$	!+IMS mode+! =$Norm$	!+IMS mode+! =$Iner$	!+IMS mode+! =$Mag sl$	!+IMS mode+! =$Grid$	!+SINS enabled+!	!+SINS velocity test passed+!	!+land velocity test passed+!	!+No intervening takeoff+!	!+ground tests finished+!	New Mode
PolarI	-	-	-	-	f	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	t	-	*Landaln*
	@T	-	-	-	f	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	
	@T	-	-	-	t	-	-	-	-	-	-	-	t	-	-	-	-	t	-	-	-	-	*SINSaln*
	-	-	-	-	f	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	t	f	-	*01Update*
	-	-	-	@T	f	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*HUDaln*
	@T	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	-	*OLB*
	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	
	@T	-	-	-	t	-	-	-	-	-	-	-	t	-	-	-	-	f	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	*Mag_sl*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	*Grid*
	-	-	-	-	-	-	-	-	-	f	@F	-	-	-	-	-	-	-	-	-	-	-	*IMS_fail*
	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	*Grtest*
Grtest	-	-	-	-	f	-	-	-	-	-	t	-	t	-	-	-	-	-	-	-	-	@T	*Landaln*
	-	-	-	-	-	-	-	-	-	-	t	-	-	t	-	-	-	-	-	-	-	@T	*OLB*
	-	-	-	-	-	-	-	-	-	-	t	-	-	-	t	-	-	-	-	-	-	@T	
	-	-	-	-	t	-	-	-	-	-	t	-	t	-	-	-	-	-	-	-	-	@T	
	-	@T	-	-	-	-	-	-	-	-	t	-	-	-	t	-	-	-	-	-	-	-	
	-	@T	-	-	-	-	-	-	-	-	t	-	-	t	-	-	-	-	-	-	-	-	
	-	@T	-	-	-	-	-	-	-	-	t	-	t	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	t	-	-	-	-	-	@T	*Mag_sl*
	-	@T	-	-	-	-	-	-	-	-	t	-	-	-	-	t	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	t	-	-	-	-	@T	*Grid*
	-	@T	-	-	-	-	-	-	-	-	t	-	-	-	-	-	t	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	-	-	-	-	-	-	@T	*IMS_fail*
	-	@T	-	-	-	-	-	-	-	-	f	-	-	-	-	-	-	-	-	-	-	-	

TABLE 131. Transitions between Alignment, Navigation, and Test Modes while the aircraft is not airborne (!+in flight+! = $true$)

Current Mode	!+new posn entered+!	!+in flight+!	!+land based panel+! = $false$	!+CA2 stage complete+!	!+CL2 stage complete+!	!+IMS up+!	!+latitude+! > 70	!+latitude+! > 80	!+Doppler up+!	!+Doppler coupled+!	!+IMS mode+! = $Gndal$	!+IMS mode+! = $Norm$	!+IMS mode+! = $Iner$	!+IMS mode+! = $Mag sl$	!+IMS mode+! = $Grid$	!+air velocity test passed$	!+pitch small+! AND !+roll small+!	New Mode
Airaln	@T	-	-	-	-	-	f	-	t	-	-	-	-	-	-	-	-	*Airaln*
	f	-	-	-	-	-	f	-	t	-	-	-	f	-	-	@T	-	*DIG*
	-	-	-	-	-	-	t	f	-	t	-	-	-	-	-	@T	-	*DI*
	-	-	-	-	t	-	-	f	-	t	-	-	@T	-	-	-	-	
	-	@F	-	-	t	-	-	f	-	-	-	-	-	-	-	-	-	*I*
	-	-	-	-	t	-	-	f	@F	-	-	-	-	-	-	-	-	
	-	-	-	-	t	-	-	f	-	f	-	-	@T	-	-	-	-	
	-	-	-	-	f	-	-	-	-	t	-	-	@T	-	-	-	t	*UDI*
	-	@F	-	-	f	-	-	-	-	-	-	-	-	-	-	-	-	*OLB*
	-	-	-	-	f	-	-	-	@F	-	-	-	-	-	-	-	-	
	-	-	-	-	f	-	-	-	-	f	-	-	@T	-	-	-	-	

TABLE 132. Transitions between Alignment, Navigation, and Test Modes while the aircraft is not airborne (!+in flight+! = $true$)

Current Mode	!+new posn entered+!	!+in flight+!	!+laInd based panel+! = $false$	!+CA2 stage complete+!	!+CL2 stage complete+!	!+IMS up+!	!+latitude+! > 70	!+latitude+! > 80	!+Doppler up+!	!+Doppler coupled+!	!+IMS mode+! = $Gndal$	!+IMS mode+! = $Norm$	!+IMS mode+! = $Iner$	!+IMS mode+! = $Mag sl$	!+IMS mode+! = $Grid$	!+air velocity test passed$	!+pitch small+! AND !+roll small+!	New Mode
DIG	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*Airaln*
	-	-	-	-	-	-	-	-	t	t	-	-	@T	-	-	-	-	*DI*
	f	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	
	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*I*
	-	-	-	-	-	-	f	-	@F	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	f	-	-	@T	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	*Mag sl*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	*Grid*
	-	-	-	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	*IMS fail*
DI	-	-	-	f	-	-	-	f	-	-	-	@T	-	-	-	-	-	*Airaln*
	-	-	-	f	-	-	-	f	-	-	@T	-	-	-	-	-	-	
	@T	-	-	-	-	-	-	-	t	-	-	t	-	-	-	-	-	
	@T	-	-	-	-	-	-	-	t	-	t	-	-	-	-	-	-	
	-	-	-	t	-	-	f	-	-	-	-	@T	-	-	-	-	-	*DIG*
	-	-	-	t	-	-	f	-	-	-	@T	-	-	-	-	-	-	
	f	-	-	t	-	-	@F	-	t	-	-	t	-	-	-	-	-	
	f	-	-	t	-	-	@F	-	t	-	t	-	-	-	-	-	-	
	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*I*
	-	-	-	-	-	-	-	-	@F	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	@F	-	-	t	-	-	-	-	
	-	-	-	-	-	-	-	-	-	f	-	-	@T	-	-	-	-	
	@T	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	t	*UDI*
	@T	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	f	*OLB*
	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	*Mag sl*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	*Grid*
	-	-	-	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	*IMS fail*
	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	*PolarDI*

TABLE 133. Transitions between Alignment, Navigation, and Test Modes while the aircraft is not airborne (!+in flight+! = $true$)

Current Mode	!+new posn entered+!	!+in flight+!	!+laInd based panel+! = $false$	!+CA2 stage complete+!	!+CL2 stage complete+!	!+IMS up+!	!+latitude+! > 70	!+latitude+! > 80	!+Doppler up+!	!+Doppler coupled+!	!+IMS mode+! = $GndaI$	!+IMS mode+! = $Norm$	!+IMS mode+! = $Iner$	!+IMS mode+! = $Mag sl$	!+IMS mode+! = $Grid$	!+air velocity test passed$	!+pitch small+! AND !+roll small+!	New Mode
I	-	@F	f	-	-	-	-	-	-	-	t	-	-	-	-	-	-	*Landaln*
	-	-	-	f	-	-	-	-	@T	-	t	-	-	-	-	-	-	*Airaln*
	-	-	-	f	-	-	-	-	@T	-	-	t	-	-	-	-	-	
	-	-	-	f	-	-	-	-	t	-	@T	-	-	-	-	-	-	
	-	-	-	f	-	-	-	-	t	-	-	@T	-	-	-	-	-	
	f	-	-	t	-	-	f	-	@T	-	t	-	-	-	-	-	-	*DIG*
	f	-	-	t	-	-	f	-	@T	-	-	t	-	-	-	-	-	
	-	-	-	t	-	-	f	-	t	-	@T	-	-	-	-	-	-	
	-	-	-	t	-	-	f	-	t	-	-	@T	-	-	-	-	-	
	-	-	-	t	-	-	t	f	t	-	@T	-	-	-	-	-	-	*DI*
	-	-	-	t	-	-	t	f	t	-	-	@T	-	-	-	-	-	
	f	-	-	-	-	-	t	f	@T	-	t	-	-	-	-	-	-	
	f	-	-	-	-	-	t	f	@T	-	-	t	-	-	-	-	-	
	f	-	-	-	-	-	-	f	@T	t	-	-	t	-	-	-	-	
	f	-	-	-	-	-	-	f	t	@T	-	-	t	-	-	-	-	
	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*OLB*
	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	*Mag_sl*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	*Grid*
	-	-	-	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	*IMS_fail*
	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	*PolarI*
UDI	-	-	-	-	-	-	-	-	t	-	-	@T	-	-	-	-	-	*Airaln*
	-	-	-	-	-	-	-	-	t	-	@T	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	-	@F	*OLB*
	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	@F	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	@F	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	*Mag_sl*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	*Grid*
	-	-	-	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	*IMS_fail*

TABLE 134. Transitions between Alignment, Navigation, and Test Modes while the aircraft is not airborne (!+in flight+! = $true$)

Current Mode	!+new posn entered+!	!+in flight+!	!+laInd based panel+! = $false$	!+CA2 stage complete+!	!+CL2 stage complete+!	!+IMS up+!	!+latitude+! > 70	!+latitude+! > 80	!+Doppler up+!	!+Doppler coupled+!	!+IMS mode+! = $Gndal$	!+IMS mode+! = $Norm$	!+IMS mode+! = $Iner$	!+IMS mode+! = $Mag sl$	!+IMS mode+! = $Grid$	!+air velocity test passed$	!+pitch small+! AND !+roll small+!	New Mode
OLB	-	-	-	-	-	-	-	-	@T	-	t	-	-	-	-	-	-	*Airaln*
	-	-	-	-	-	-	-	-	@T	-	-	t	-	-	-	-	-	
	-	-	-	-	-	-	-	-	t	-	@T	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	t	-	-	@T	-	-	-	-	-	
	-	-	-	-	-	-	-	-	@T	t	-	-	t	-	-	-	t	*UDI*
	-	-	-	-	-	-	-	-	t	@T	-	-	t	-	-	-	t	
	-	-	-	-	-	-	-	-	t	t	-	-	-	-	-	-	@T	
	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	*Mag_sl*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	*Grid*
	-	-	-	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	*IMS_fail*
Mag_sl	-	-	-	-	-	-	-	-	t	-	@T	-	-	-	-	-	-	*Airaln*
	-	-	-	-	-	-	-	-	t	-	-	@T	-	-	-	-	-	
	-	-	-	-	-	-	-	-	t	t	-	-	@T	-	-	-	t	*UDI*
	-	-	-	-	-	-	-	-	f	-	-	-	@T	-	-	-	-	*OLB*
	-	-	-	-	-	-	-	-	-	f	-	-	@T	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	f	
	-	-	-	-	-	-	-	-	f	-	@T	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	f	-	-	@T	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	*Grid*
	-	-	-	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	*IMS_fail*

TABLE 135. Transitions between Alignment, Navigation, and Test Modes while the aircraft is not airborne (!+in flight+! = $true$)

Current Mode	!+new posn entered+!	!+in flight+!	!+land based panel+! = $false$	!+CA2 stage complete+!	!+CL2 stage complete+!	!+IMS up+!	!+latitude+! > 70	!+latitude+! > 80	!+Doppler up+!	!+Doppler coupled+!	!+IMS mode+! = $Gndal$	!+IMS mode+! = $Norm$	!+IMS mode+! = $Iner$	!+IMS mode+! = $Mag sl$	!+IMS mode+! = $Grid$	!+air velocity test passed$	!+pitch small+! AND !+roll small+!	New Mode
Grid	-	-	-	-	-	-	-	-	t	-	@T	-	-	-	-	-	-	*Airaln*
	-	-	-	-	-	-	-	-	t	-	-	@T	-	-	-	-	-	
	-	-	-	-	-	-	-	-	t	t	-	-	@T	-	-	-	t	*UDI*
	-	-	-	-	-	-	-	-	f	-	-	-	@T	-	-	-	-	*OLB*
	-	-	-	-	-	-	-	-	-	f	-	-	@T	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	f	
	-	-	-	-	-	-	-	-	f	-	@T	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	f	-	-	@T	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	*Mag_sl*
	-	-	-	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	*IMS_fail*
IMS_fail	-	-	-	-	-	@T	-	-	t	-	t	-	-	-	-	-	-	*Airaln*
	-	-	-	-	-	@T	-	-	t	-	-	t	-	-	-	-	-	
	-	-	-	-	-	@T	-	-	t	t	-	-	t	-	-	-	t	*UDI*
	-	-	-	-	-	@T	-	-	f	-	-	-	t	-	-	-	-	*OLB*
	-	-	-	-	-	@T	-	-	-	f	-	-	t	-	-	-	-	
	-	-	-	-	-	@T	-	-	-	-	-	-	t	-	-	-	f	
	-	-	-	-	-	@T	-	-	f	-	t	-	-	-	-	-	-	
	-	-	-	-	-	@T	-	-	f	-	-	t	-	-	-	-	-	
	-	-	-	-	-	@T	-	-	-	-	-	-	-	t	-	-	-	*Mag_sl*
	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	t	-	-	*Grid*

TABLE 136. Transitions between Alignment, Navigation, and Test Modes while the aircraft is not airborne (!+in flight+! = $true$)

Current Mode	!+new posn entered+!	!+in flight+!	!+land based panel+! = $false$	!+CA2 stage complete+!	!+CL2 stage complete+!	!+IMS up+!	!+latitude+! > 70	!+latitude+! > 80	!+Doppler up+!	!+Doppler coupled+!	!+IMS mode+! = $Gndal$	!+IMS mode+! = $Norm$	!+IMS mode+! = $Iner$	!+IMS mode+! = $Mag sl	!+IMS mode+! = $Grid$	!+air velocity test passed$	!+pitch small+! AND !+roll small+!	New Mode
PolarDI	@T	-	-	-	-	-	-	-	t	-	-	t	-	-	-	-	-	*Airaln*
	@T	-	-	-	-	-	-	-	t	-	t	-	-	-	-	-	-	
	@T	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	t	*UDI*
	@T	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	f	*OLB*
	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	*Mag sl*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	*Grid*
	-	-	-	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	*IMS_fail*
	-	-	-	-	-	-	-	-	@F	-	-	-	-	-	-	-	-	*PolarI*
	-	-	-	-	-	-	-	-	-	@F	-	-	t	-	-	-	-	
	-	-	-	-	-	-	-	-	-	f	-	-	@T	-	-	-	-	
	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
PolarI	-	@F	f	-	-	-	-	-	-	-	t	-	-	-	-	-	-	*Landaln*
	-	-	-	f	-	-	-	-	@T	-	t	-	-	-	-	-	-	*Airaln*
	-	-	-	f	-	-	-	-	@T	-	-	t	-	-	-	-	-	
	-	-	-	f	-	-	-	-	t	-	@T	-	-	-	-	-	-	
	-	-	-	f	-	-	-	-	t	-	-	@T	-	-	-	-	-	
	@T	-	-	-	-	-	-	-	t	-	t	-	-	-	-	-	-	
	@T	-	-	-	-	-	-	-	t	-	-	t	-	-	-	-	-	
	@T	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	*OLB*
	@T	-	-	-	-	-	-	-	f	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	*Mag_sl*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	*Grid*
	-	-	-	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	*IMS_fail*
	-	-	-	-	-	-	-	-	@T	t	-	-	t	-	-	-	-	*PolarDI*
	-	-	-	t	-	-	-	-	@T	-	t	-	-	-	-	-	-	
	-	-	-	t	-	-	-	-	@T	-	-	t	-	-	-	-	-	
	-	-	-	-	-	-	-	-	t	@T	-	t	-	-	-	-	-	
	-	-	-	t	-	-	-	-	t	-	@T	-	-	-	-	-	-	
	-	-	-	t	-	-	-	-	t	-	-	@T	-	-	-	-	-	

16.2 Navigation Update Modes

The initial navigation update mode is *Unone*.

TABLE 137. Transitions between Navigation Update Modes

Current Mode	!+Station selected+!	!+in flight+!	!+panel mode+! = $Prespos$	!+pres pos+! = $Update$	!+Update+! = $Flyover$	!+Update+! = HUD	!+Update+! = $Radar$	!+Update+! = $TacL-L$	!+UpdATTW=Other+!	!+Gun Enable+!	!+WDMFS+!	!+land based pnl+! = $false$	!+in BOC+!	NOT !+boc+! & NOT !+ccip+! & NOT !+natt+! & NOT !+tf+! & NOT !+offset+!	!+tf+!	!+keybd input+! != 0 & !+keybd input+! != $None$	!+Enter pressed+!	!+Fly to num changed+! OR !+Fly to state changed+!	New Mode
UNone	f	t	t	t	-	t	-	-	-	f	-	-	-	t	-	@T	-	-	*HUDUpd*
	f	t	t	t	-	-	t	-	-	f	-	-	-	t	-	@T	-	-	*RadarUpd*
	f	f	t	t	-	-	t	-	-	f	-	f	-	t	-	@T	-	-	
	-	t	t	t	t	-	-	-	-	-	-	-	t	-	-	@T	-	-	*AflyUpd*
	@F	-	t	t	t	-	-	-	-	f	f	-	-	-	-	-	-	-	*MapUpd*
	f	-	@T	t	t	-	-	-	-	f	f	-	-	-	-	-	-	-	
	f	-	t	@T	t	-	-	-	-	f	f	-	-	-	-	-	-	-	
	f	-	t	t	@T	-	-	-	-	f	f	-	-	-	-	-	-	-	
	f	-	t	t	t	-	-	-	-	@F	f	-	-	-	-	-	-	-	
	f	-	t	t	t	-	-	-	-	f	@F	-	-	-	-	-	-	-	
	f	t	t	t	-	-	-	t	-	f	f	-	-	-	-	@T	-	-	*TacUpd*
	t	t	t	t	-	-	-	t	-	f	f	-	-	-	t	@T	-	-	
HUDUpd	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*UNone*
	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	
	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	
	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	*HUDUpd*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	
	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	*MapUpd*

TABLE 138. Transitions between Navigation Update Modes

Current Mode	!+Station selected+!	!+in flight+!	!+panel mode+! = $Prespos$	!+pres pos+! = $Update$	!+Update+! = $Flyover$	!+Update+! = HUD	!+Update+! = $Radar$	!+Update+! = $TacL-L$	!+UpdATTW=Other+!	!+Gun Enable+!	!+WDMFS+!	!+land based pnl+! = $false$	!+in BOC+!	NOT !+boc+! & NOT !+ccip+! & NOT !+natt+! & NOT !+tf+! & NOT !+offset+!	!+tf+!	!+keybd input+! != 0 & !+keybd input+! != $None$	!+Enter pressed+!	!+Fly to num changed+! OR !+Fly to state changed+!	New Mode
RadarUpd	-	@F	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	-	*UNone*
	-	f	-	-	-	-	-	-	-	-	-	@F	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	
	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	
	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	*RadarUpd*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	
	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	*MapUpd*
FlyUpd	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	*UNone*
	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	
	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	*MapUpd*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	

TABLE 139. Transitions between Navigation Update Modes

Current Mode	!+Station selected+!	!+in flight+!	!+panel mode+! = $Prespos$	!+pres pos+! = $Update$	!+Update+! = $Flyover$	!+Update+! = HUD	!+Update+! = $Radar$	!+Update+! = $TacL-L$	!+UpdATTW=Other+!	!+Gun Enable+!	!+WDMFS+!	!+land based pnl+! = $false$	!+in BOC+!	NOT !+boc+! & NOT !+ccip+! & NOT !+natt+! & NOT !+tf+! & NOT !+offset+!	!+tf+!	!+keybd input+! != 0 & !+keybd input+! != $None$	!+Enter pressed+!	!+Fly to num changed+! OR !+Fly to state changed+!	New Mode
AflyUpd	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*UNone*
	-	-	-	-	-	-	-	-	-	-	-	-	@F	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	
	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	
MapUpd	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	*UNone*
	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	
	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	-	*FlyUpd*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	*MapUpd*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	
TacUpd	-	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	*UNone*
	-	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	
	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	@F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	@T	-	
	-	-	-	-	@T	-	-	-	-	-	-	-	-	-	-	-	-	-	*MapUpd*

16.3 Weapon Delivery Modes

The initial weapon delivery mode is *Wnone*.

TABLE 140. Transitions between Weapon Delivery Modes

Current Mode	!+Ready Station REQ+!	!+HUD reliable+!	!+Reserved Weapon+!	!+Special+!	!+Rockets+!	!+Guns+!	!+Walleye+!	!+Shrike+!	!+Other weapon+!	NOT !+Gun Enable+!	!+Fly to num+! = 0	!+fly to num reset+!	!+Fly to State+! = $Dest$	!+WDMFS+!	!+boc+!	!+boc+! & +!offset+!	!+ccip+!	!+natt+!	!+natt+! & !+offset+!	!+desig+!	!+Redesignate+!	!+new dest coords entered+!	!+high drag release+!	!+low drag release+!	!+Overflown exit+!	!+gr ac stik exit+! > 42 nmi.	!+in flight+!	!+in Off_MFSW+!	!+in WD_MFSW+!	New Mode
WNone	-	-	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	OFF_MFS
	-	-	-	-	-	-	-	-	-	T	-	-	-	f	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	
	-	-	-	T	-	-	-	-	-	-	-	-	-	f	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	
	-	-	-	-	T	-	-	-	-	-	-	-	-	f	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	
	-	-	-	-	-	T	-	-	-	-	-	-	-	f	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	
	-	-	-	-	-	-	T	-	-	-	-	-	-	f	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	
	-	-	-	-	-	-	-	T	-	-	-	-	-	f	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	
	-	-	-	-	-	-	-	-	T	-	-	-	-	f	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	WD_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	t	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	t	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	t	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	t	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	t	-	-	
	F	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	
	-	-	-	-	T	-	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	
	-	-	-	-	-	-	-	T	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	
	-	-	-	-	-	-	-	-	T	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	
	-	-	-	-	-	-	-	-	-	-	T	-	t	t	-	-	-	-	-	-	-	-	-	-	-	f	-	t	-	-
	-	-	-	-	-	-	-	-	-	-	t	-	T	t	-	-	-	-	-	-	-	-	-	-	-	f	-	t	-	-
	-	T	-	t	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
OFF_MFS	-	-	-	t	-	-	-	-	-	f	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	*WNone*
	-	-	-	t	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	*A/A_Guns*
	f	-	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	
	t	-	-	f	-	-	-	-	-	f	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	*Manrip*
	t	-	-	f	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	*A/A_Manrip*

TABLE 141. Transitions between Weapon Delivery Modes

Current Mode	!+Ready Station REQ+!	!+HUD reliable+!	!+Reserved Weapon+!	!+Special+!	!+Rockets+!	!+Guns+!	!+Walleye+!	!+Shrike+!	!+Other weapon+!	NOT !+Gun Enable+!	!+Fly to num+! = 0	!+fly to num reset+!	!+Fly to State+! = $Dest$	!+WDMFS+!	!+boc+!	!+boc+! & +!offset+!	!+ccip+!	!+natt+!	!+natt+! & !+offset+!	!+desig+!	!+Redesignate+!	!+new dest coords entered+!	!+high drag release+!	!+low drag release+!	!+Overflown exit+!	!+gr ac stik exit+! > 42 nmi.	!+in flight+!	!+in Off_MFSW+!	!+in WD_MFSW+!	New Mode
WD_MFS	-	-	-	-	-	-	-	-	-	-	-	-	f	-	-	t	-	-	-	-	-	-	-	-	-	-	-	-	T	*WNone*
	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	-	-	T	
	-	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	T	
	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	T	
	-	-	t	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	-	T	
	-	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	-	-	-	-	-	-	T	
	-	f	-	t	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	T	
	t	t	f	-	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	T	*Nattack*
	t	t	f	-	-	-	-	-	-	-	t	-	t	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	T	*Noffset*
	t	-	f	-	-	-	-	-	-	-	f	-	-	-	t	-	-	-	-	-	-	-	-	-	-	-	-	-	T	*BOC*
	t	-	f	-	-	-	-	-	-	-	t	-	-	-	t	-	-	-	-	-	-	-	-	-	-	-	-	-	T	*BOCFlyto0*
	t	-	f	-	-	-	-	-	-	-	t	-	t	-	-	t	-	-	-	-	-	-	-	-	-	-	-	-	T	*BOCoffset*
	t	-	f	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	-	T	*CCIP*
	t	f	f	-	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	T	*HUDdown1*
	t	f	f	-	-	-	-	-	-	-	t	-	t	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	T	*HUDdown2*
	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	*A/G_Guns*
	f	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	
	t	-	-	t	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	t	-	-	-	-	-	T	*Snattack*
	t	t	-	t	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	T	
	t	-	-	t	-	-	-	-	-	-	t	-	t	-	-	-	-	-	t	-	-	-	t	-	-	-	-	-	T	*Snoffset*
	t	t	-	t	-	-	-	-	-	-	t	-	t	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	T	
	t	-	-	t	-	-	-	-	-	-	f	-	-	-	t	-	-	-	-	-	-	-	t	-	-	-	-	-	T	*SBOC*
	t	t	-	t	-	-	-	-	-	-	f	-	-	-	t	-	-	-	-	-	-	-	-	-	-	-	-	-	T	
	t	-	-	t	-	-	-	-	-	-	t	-	-	-	t	-	-	-	-	-	-	-	t	-	-	-	-	-	T	*SBOCFlyto0*
	t	t	-	t	-	-	-	-	-	-	t	-	-	-	t	-	-	-	-	-	-	-	-	-	-	-	-	-	T	
	t	-	-	t	-	-	-	-	-	-	t	-	t	-	-	t	-	-	-	-	-	-	t	-	-	-	-	-	T	*SBOCoffset*
	t	t	-	t	-	-	-	-	-	-	t	-	t	-	-	t	-	-	-	-	-	-	-	-	-	-	-	-	T	
	t	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	-	-	-	-	-	T	*Walleye*

TABLE 142. Transitions between Weapon Delivery Modes

Current Mode	!+Ready Station REQ+!	!+HUD reliable+!	!+Reserved Weapon+!	!+Special+!	!+Rockets+!	!+Guns+!	!+Walleye+!	!+Shrike+!	!+Other weapon+!	NOT !+Gun Enable+!	!+Fly to num+! = 0	!+fly to num reset+!	!+Fly to State+! = $Dest$	!+WDMFS+!	!+boc+!	!+boc+! & +!offset+!	!+ccip+!	!+natt+!	!+natt+! & !+offset+!	!+desig+!	!+Redesignate+!	!+new dest coords entered+!	!+high drag release+!	!+low drag release+!	!+Overflown exit+!	!+gr ac stik exit+! > 42 nmi.	!+in flight+!	!+in Off_MFSW+!	!+in WD_MFSW+!	New Mode
Nattack	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	T	-	-	-	*WNone*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	OFF_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WD_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	
	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	
	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	*Nattack*
	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	
	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	-	-	-	-	-	*HUDdown1*
Noffset	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	*WNone*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	T	-	-	-	
	-	-	-	-	-	-	-	-	-	-	T	-	t	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	OFF_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WD_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	
	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*Noffset*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	
	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	-	-	-	-	-	*HUDdown2*

TABLE 143. Transitions between Weapon Delivery Modes

Current Mode	!+Ready Station REQ+!	!+HUD reliable+!	!+Reserved Weapon+!	!+Special+!	!+Rockets+!	!+Guns+!	!+Walleye+!	!+Shrike+!	!+Other weapon+!	NOT !+Gun Enable+!	!+Fly to num+! = 0	!+fly to num reset+!	!+Fly to State+! = $Dest$	!+WDMFS+!	!+boc+!	!+boc+! & +!offset+!	!+ccip+!	!+natt+!	!+natt+! & !+offset+!	!+desig+!	!+Redesignate+!	!+new dest coords entered+!	!+high drag release+!	!+low drag release+!	!+Overflown exit+!	!+gr ac stik exit+! > 42 nmi.	!+in flight+!	!+in Off_MFSW+!	!+in WD_MFSW+!	New Mode
BOC	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	*WNone*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	T	-	-	
	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	OFF_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	WD_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	
	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	f	-	T	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	*BOC*
	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	*BOCFlyto0*
BOCFlyto0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	*WNone*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	T	-	-	
	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	OFF_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	WD_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	
	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*BOC*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	*BOCFlyto0*

TABLE 144. Transitions between Weapon Delivery Modes

Current Mode	!+Ready Station REQ+!	!+HUD reliable+!	!+Reserved Weapon+!	!+Special+!	!+Rockets+!	!+Guns+!	!+Walleye+!	!+Shrike+!	!+Other weapon+!	NOT !+Gun Enable+!	!+Fly to num+! = 0	!+fly to num reset+!	!+Fly to State+! = $Dest$	!+WDMFS+!	!+boc+!	!+boc+! & +!offset+!	!+ccip+!	!+natt+!	!+natt+! & !+offset+!	!+desig+!	!+Redesignate+!	!+new dest coords entered+!	!+high drag release+!	!+low drag release+!	!+Overflown exit+!	!+gr ac stik exit+! > 42 nmi.	!+in flight+!	!+in Off_MFSW+!	!+in WD_MFSW+!	New Mode
BOCoffset	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	*WNone*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	T	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	T	-	t	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	
	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	OFF_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WD_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	
	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*BOCoffset*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	
CCIP	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	*WNone*
	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	OFF_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WD_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	
	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	

TABLE 145. Transitions between Weapon Delivery Modes

Current Mode	!+Ready Station REQ+!	!+HUD reliable+!	!+Reserved Weapon+!	!+Special+!	!+Rockets+!	!+Guns+!	!+Walleye+!	!+Shrike+!	!+Other weapon+!	NOT !+Gun Enable+!	!+Fly to num+! = 0	!+fly to num reset+!	!+Fly to State+! = $Dest$	!+WDMFS+!	!+boc+!	!+boc+! & +!offset+!	!+ccip+!	!+natt+!	!+natt+! & !+offset+!	!+desig+!	!+Redesignate+!	!+new dest coords entered+!	!+high drag release+!	!+low drag release+!	!+Overflown exit+!	!+gr ac stik exit+! > 42 nmi.	!+in flight+!	!+in Off_MFSW+!	!+in WD_MFSW+!	New Mode
HUDdown1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	*WNone*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	T	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	OFF_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WD_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	
	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	T	-	f	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	-	-	-	-	-	*Nattack*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	*HUDdown1*
HUDdown2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	*WNone*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	T	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	T	-	t	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	
	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	OFF_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WD_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	
	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	T	-	f	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	-	-	-	-	-	*Noffset*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	*HUDdown2*

TABLE 146. Transitions between Weapon Delivery Modes

Current Mode	!+Ready Station REQ+!	!+HUD reliable+!	!+Reserved Weapon+!	!+Special+!	!+Rockets+!	!+Guns+!	!+Walleye+!	!+Shrike+!	!+Other weapon+!	NOT !+Gun Enable+!	!+Fly to num+! = 0	!+fly to num reset+!	!+Fly to State+! = $Dest$	!+WDMFS+!	!+boc+!	!+boc+! & +!offset+!	!+ccip+!	!+natt+!	!+natt+! & !+offset+!	!+desig+!	!+Redesignate+!	!+new dest coords entered+!	!+high drag release+!	!+low drag release+!	!+Overflown exit+!	!+gr ac stik exit+! > 42 nmi.	!+in flight+!	!+in Off_MFSW+!	!+in WD_MFSW+!	New Mode
A/G_Guns	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	*WNone*
	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	OFF_MFS
	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WD_MFS
	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
A/A_Guns	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	*WNone*
	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	OFF_MFS
	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WD_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	
Manrip	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	*WNone*
	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	OFF_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WD_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	

TABLE 147. Transitions between Weapon Delivery Modes

Current Mode	!+Ready Station REQ+!	!+HUD reliable+!	!+Reserved Weapon+!	!+Special+!	!+Rockets+!	!+Guns+!	!+Walleye+!	!+Shrike+!	!+Other weapon+!	NOT !+Gun Enable+!	!+Fly to num+! = 0	!+fly to num reset+!	!+Fly to State+! = $Dest$	!+WDMFS+!	!+boc+!	!+boc+! & +!offset+!	!+ccip+!	!+natt+!	!+natt+! & !+offset+!	!+desig+!	!+Redesignate+!	!+new dest coords entered+!	!+high drag release+!	!+low drag release+!	!+Overflown exit+!	!+gr ac stik exit+! > 42 nmi.	!+in flight+!	!+in Off_MFSW+!	!+in WD_MFSW+!	New Mode
A/A Manrip	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	*WNone*
	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	OFF_MFS
	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WD_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	
Snattack	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	*WNone*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	T	-	-	-	
	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WD_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*Snattack*
	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	

TABLE 148. Transitions between Weapon Delivery Modes

Current Mode	!+Ready Station REQ+!	!+HUD reliable+!	!+Reserved Weapon+!	!+Special+!	!+Rockets+!	!+Guns+!	!+Walleye+!	!+Shrike+!	!+Other weapon+!	NOT !+Gun Enable+!	!+Fly to num+! = 0	!+fly to num reset+!	!+Fly to State+! = $Dest$	!+WDMFS+!	!+boc+!	!+boc+! & +!offset+!	!+ccip+!	!+natt+!	!+natt+! & !+offset+!	!+desig+!	!+Redesignate+!	!+new dest coords entered+!	!+high drag release+!	!+low drag release+!	!+Overflown exit+!	!+gr ac stik exit+! > 42 nmi.	!+in flight+!	!+in Off_MFSW+!	!+in WD_MFSW+!	New Mode
Snoffset	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	*WNone*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	T	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	T	-	t	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	
	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WD_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*Snoffset*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	
SBOC	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	*WNone*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	T	-	-	-	
	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	WD_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	f	-	T	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	*SBOC*
	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	*SBOCFlyto0*

TABLE 149. Transitions between Weapon Delivery Modes

Current Mode	!+Ready Station REQ+!	!+HUD reliable+!	!+Reserved Weapon+!	!+Special+!	!+Rockets+!	!+Guns+!	!+Walleye+!	!+Shrike+!	!+Other weapon+!	NOT !+Gun Enable+!	!+Fly to num+! = 0	!+fly to num reset+!	!+Fly to State+! = $Dest$	!+WDMFS+!	!+boc+!	!+boc+! & +!offset+!	!+ccip+!	!+natt+!	!+natt+! & !+offset+!	!+desig+!	!+Redesignate+!	!+new dest coords entered+!	!+high drag release+!	!+low drag release+!	!+Overflown exit+!	!+gr ac stik exit+! > 42 nmi.	!+in flight+!	!+in Off_MFSW+!	!+in WD_MFSW+!	New Mode
SBOCFlyto0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	*WNone*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	T	-	-	-	
	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	WD_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*SBOC*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	*SBOCFlyto0*
	-	-	-	-	-	-	-	-	-	-	t	-	T	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
SBOCoffset	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	*WNone*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-	-	-	-	T	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	T	-	t	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	
	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WD_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	*SBOCoffset*
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	

TABLE 150. Transitions between Weapon Delivery Modes

Current Mode	!+Ready Station REQ+!	!+HUD reliable+!	!+Reserved Weapon+!	!+Special+!	!+Rockets+!	!+Guns+!	!+Walleye+!	!+Shrike+!	!+Other weapon+!	NOT !+Gun Enable+!	!+Fly to num+! = 0	!+fly to num reset+!	!+Fly to State+! = $Dest$	!+WDMFS+!	!+boc+!	!+boc+! & +!offset+!	!+ccip+!	!+natt+!	!+natt+! & !+offset+!	!+desig+!	!+Redesignate+!	!+new dest coords entered+!	!+high drag release+!	!+low drag release+!	!+Overflown exit+!	!+gr ac stik exit+! > 42 nmi.	!+in flight+!	!+in Off_MFSW+!	!+in WD_MFSW+!	New Mode
Walleye	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	*WNone*
	-	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	OFF_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	WD_MFS
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	
	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	*Walleye*
	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	T	-	-	-	-	-	-	-	-	-	-	-	f	-	-	-	-	
	-	-	-	-	-	-	-	-	-	-	-	-	F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	

17.0 Type Dictionary

TABLE 151. Type Dictionary (Sheet 1 of 2)

Type Name	Values
accel	
angle	
angrate	
astage	Enumerated: CA, $CA2$, CL, $CL2$, ED, $ED2$, FM, FG, $FG2$, HG, HL, HS, ND, $ND2$, TS, $TS2$, $None$.
AUD_ind_cntrl	Enumerated: On, Off, $Intermittent$
boolean	$true$, $false$, or true, false
boolean array	
char	Enumerated: $blank$, $hyphen$, or x where x is in {ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789}.
char_string_*n*	An array of n elements of type char. The lower bound is assumed to be zero; the upper bound n-1.
distance	
drs_mode	Enumerated: Off,$Operate$, $Memory$, $Stndby$, $Test$
event	
FLR_ind_cntrl	Enumerated: On, Off, $Intermittent$
flr_mode	Enumerated: $Ranging$, $Idle$, $CDCE$, TF
fly_to_state	Enumerated: $Dest$ $Mark$
format	NONE, ANGLE, CHAR_UINT, CHARSTR, LATITUDE, LONGITUDE, REAL, SFRAC, SIGN, SINT, UFRAC, TIME, UINT, CLEAR, BLNKLTS,
HUD_ind_cnrtl	Enumerated: On, Off, $Intermittent$
HUD_test	Enumerated: A, B, $None$
imsmode	Enumerated: $Gndal$, $Norm$, $Iner$, $Magsl$, $Grid$, Off
imsscale	Simple enumerated: $Fine$, $Coarse$
integer	A member of the real typeclass that has the EXACT_REP attribute and whose !!resolution!! = 1.
keybd	Enumerated: $None$, 0, 1, $N2$, $L3$, $W4$, $H5$, $C7$, $S8$, $-E6$, $D9$, $Error$.
latitude	An angle whose range goes from -90 to +90, inclusively. A negative (positive) value represents a West (East) latitude.
longitude	An angle whose range goes from -180 to +180, inclusively. A negative (positive) value represents a South (North) longitude.
map_scale	Enumerated. The values of the map scale data type have a complete ordering from finest to grossest, where a map at finest scale shows the most detail and at grossest scale shows the largest area.

TABLE 151. Type Dictionary (Sheet 2 of 2)

Type Name	Values
panel_config	$align stage$, $alt AGL at rls$, $alt baro AGL$, $ARPINT$, $ARPQUANT$, $az miss dist at rls$, $az ref hdg$, $burst ht$, $central long a$, $central long b$, $compfail$, $data nbr$, $dest altitude$, $dest lat$, $dest long$, $dest mslp$, $Doppler coupled$, $drftangl IMS$, $drift angle filtered$, $e coarse bias$, $e coarse scale$, $e fine bias$, $e fine scale$, $elapsed navaln time$, $fpangl at rls$, $gndspd filtered$, $groundspeed IMS$, $gyro drift delta n$, $hdg system$, $heading IMS$, $heading MAG$, $IMS diags1$, $IMS diags2$, $IMS total vel$, $L-probe$, $land based$, $latitude error$, $latitude$, $longitude error$, $longitude$, $low lat ct a$, $low lat ct b$, $mag variation$, $map latitude$, $map longitude$, $map orient a$, $map orient b$, $map sw diags$, $mark lat$, $mark long$, $MFSW diags$, $n coarse bias$, $n coarse scale$, $n fine bias$, $n fine scale$, $nav diags1$, $nav diags2$, $none$, $norm accel at rls$, $offset brg$, $offset dht$, $offset rng$, $OFP ver1$, $OFP ver2$, $priority alt display$, $radalt priority$, $SINS dhdg$, $SINS east vel$, $SINS heading$, $SINS lat$, $SINS long$, $SINS north vel$, $SINS valid1$, $SINS valid2$, $SINS x offset$, $SINS y offset$, $SINS z offset$, $slant range at rls$, $STARDY diags$, $TAS ADC at rls$, $TAS ADC$, $time to dest$, $v coarse bias$, $v coarse scale$, $vel e$, $vel n$, $WEAPTYP$, $wind dir$, $wind speed$, $wpn sw diags$, $x corr increm$, $x drift$, $y corr increm$, $y drift$, $z corr increm$, $z drift$
panel_mode	Enumerated: $None$, $Prespos$, $Dest$, $Mark$, Rng/Brg, $DBHT$, $ALTMSLP$
pp_mode	Enumerated: $LatLong$, $Update$, $Wind$
pressure	
real	An approximation to conventional real numbers.
ref_pt	Enumerated: $center$, $bottom-center$
sensor_name	Enumerated: $None$, A, F, H.
speed	
timeint	Representation of a time interval.
tstage	Enumerated: CS, TM, GA, DIO, SC, DC, $AC1$, $AC2$, PD, $None$.
update	Enumerated: $Data$, HUD, $Radar$, $Flyover$, $Loran$, $TacL-L$, $Tacmv$, $IMS-HUD$, $SINSX-Y$, $Z-DHDG$
velocity	
VIS_ind_cntrl	Enumerated: On, Off, $Intermittent$
weap_class	GN, HD, MF, MD, OD, OR, RK, SH, SK, SL, SM, SOD, SSH, UN, WL
weap_mode	Enumerated: $NATT$, BOC, $BOCOFF$, $CCIP$, $NATTOFF$, $None$.

18.0 System Generation Parameter Dictionary

TABLE 152. System Generation Parameter Dictionary (Sheet 1 of 2)

Constant	Type	Expected Values	Value Constraints	Description
#close down time#	timeint			The minimum expected time interval between the moment that the computer enters !+failed state+! and the moment when no more software actions may occur.
#HUD symbol az max#	angle		12 deg – 20 deg.	The maximum of !+* azimuth+! where * is replaced by one of the following symbols: AS, ASL, FPM, LSC, USC and PUAC.
#HUD symbol az min#	angle		-20 deg.– -12 deg.	The minimum of !+* azimuth+! where * is replaced by one of the following symbols: AS, ASL, FPM, LSC, USC and PUAC.
#HUD symbol el res#	angle		0.005859375 deg. – 0.009765625 deg.	The resolution of !+* elevation+! where * is replaced by one of the following symbols: AS, ASL, FPM, LSC, USC and PUAC.
#IMS adj xy tolerance#	angle			
#ims cutoff#	angle	20 min. of arc		
#IMSR init coarse vscale#	speed			Initial value of the coarse scale parameters.
#IMSR init fine vscale#	speed			Initial value of the fine scale parameters.
#Map scale array#	map_scale			Array of map scale values, number of entries equals #Map scale array#. #Map scale array#(1) is the finest map scale and #Map scale array#(#Num map scales#) is the grossest map scale.
#max data nbr#	integer			The maximum value for !+data nbr panel+!. If the pilot attempts to enter a value larger than this, @T(!+panel error+! will occur.
#multval hbound#	integer			The highest legal integer associated with a multiple-value panel input item.
#multval lbound#	integer			The lowest legal integer associated with a multiple-value panel input item.
#navaln wraparound#	timeint		16,400 sec – 32,700 sec	The maximum navigation/alighment time interval measurable.

TABLE 152. System Generation Parameter Dictionary (Sheet 2 of 2)

Constant	Type	Expected Values	Value Constraints	Description
#Num map scales#	integer	10		Number of map scales available.
#num weap stations#	integer		0 – 8	The num of Weapon stations controlled by the software. Note that the aircraft may have some weapon staton(s) that cannot be controlled by the software.
#OFP version lower#	char_string(7)	"OFP-A7"		
#OFP version upper#	char_string(6)	"NRL-1"		
#weap interval max#	integer		750 – 1250	The miximum setting of the interval switch.
#weap quantity max#	integer		75 – 125	The maximum setting of the quantity switch.

19.0 Variable Dictionary

TABLE 153. Variable Dictionary (Sheet 1 of 22)

MCT	Name	Type	Interpretation
T	!+a/c location+!		The present latitude and longitude of the aircraft.
M	!+AC test results+!	boolean	true iff the AC signal converter passes built-in test.
M	!+adc alt up+!	boolean	true iff the Air Data Computer is functioning and producing current and reasonable altitude readings.
M	!+adc reasonable+!	boolean	true iff the Air Data Computer is producing reasonable results for at least some of its reported values.
M	!+adc tas up+!	boolean	true iff the Air Data Computer is functioning and producing current and reasonable true airspeed readings.
M	!+ADCFAIL+!	boolean	The current value of the EC i/o item /ADCFAIL/.
C	!+ADI az+!	angle	The value currently being displayed on the ADI azimuth display. Positive if the azimuth of the display is to the right (as viewed by the pilot) of the ADI reference point, negative otherwise.
M	!+ADI elev avail+!	boolean	$true$ iff that the ADI elevation indicator is available for software control.
C	!+ADI elev in view+!	boolean	$true$ iff the ADI elevation indicator is displayed in view of the pilot.
C	!+ADI elev+!	angle	The value currently being displayed by the ADI elevation display, or (if it is out of view) the value that would be displayed were it in view. Positive if the elevation of the display is above (as viewed by the pilot) of the ADI reference point, negative otherwise.
M	!+after slewing+!	boolean	true iff legal slew inputs have been entered at least once since entry into the current mode.
T	!+aiming switches+!	boolean	true iff !+DI.Panel mode+!=$Prespos$ AND !+DI.Pres pos+!=$Update$ AND !+DI.Update+!=$IMS-HUD$.
M	!+air velocity test passed+!	boolean	true iff the difference between Doppler- and IMS-measured velocities is within acceptable bounds.
T	!+align stage eq *x*+!	boolean	True iff !+align stage+! = $*x*$.
M	!+align stage+!	astage	The current alignment mode stage of the system. Value is $None$ if the system is in no alignment stage.
M	!+alt ADC+!	distance	If !+alt ADC valid+! then !+alt ADC+! is the ADC-measured altitude of the aircraft above sea level. Otherwise, !+alt ADC+! is undefined.
M	!+alt ADC valid+!	boolean	If !+alt ADC valid+! then !+alt ADC+! is the ADC-measured altitude of the aircraft above sea level. Otherwise, !+alt ADC+! is undefined.
M	!+alt priority ranging+!	distance	The current altitude of the aircraft, from the best available sensors, when in any weapons modes except $wnone$, $a/g guns$, $walleye$, and $a/a guns$. In those modes, it is undefined. This altitude may be either above sea level, if the best sensor available is barometric, or above ground level, if the best sensor is the FLR or Radar Altimeter.
M	!+alt priority source+!	sensor_name	The sensor from which the current value of !+alt priority stale+! was obtained. If $None$, then the corresponding !+alt priority stale+! will be zero.

TABLE 153. Variable Dictionary (Sheet 2 of 22)

MCT	Name	Type	Interpretation
M	!+alt priority stale+!	distance	The value !+alt priority ranging+! frozen at some specified moment in the recent past, or zero. When the value is updated is determined by certain events and mode transitions during the flight.
M	!+AOA valid+!	boolean	$true$ indicates !+AOA+! is current. $false$ indicates !+AOA+! may not accurately reflect current aircraft angle of attack.
M	!+AOA+!	angle	Current aircraft angle of attack.
MC	!+AS azimuth+!	angle	Present angle between the ORA and the projection of the ray between the HUD origin and the symbol onto the plane formed by the ORA and the !!horizontal ORA!! or (if the symbol is currently off) the azimuth angle at which the symbol would be displayed. The angle is positive (negative) if to the right (left) of the positive ORA as viewed by the pilot.
MC	!+AS elevation+!	angle	Present angle between the ORA and the projection of the ray between the HUD origin and the symbol onto the plane formed by the ORA and the !!vertical ORA!! or (if the symbol is currently off) the elevation angle at which the symbol would be displayed. The angle is positive (negative) if to the above (below) of the positive ORA as viewed by the pilot.
C	!+AS mode+!	HUD_ind_cnrtl	The display mode of the HUD aiming symbol.
C	!+ASL azimuth+!	angle	Present angle between the ORA and the projection of the ray between the HUD origin and the symbol onto the plane formed by the ORA and the !!horizontal ORA!! or (if the symbol is currently off) the azimuth angle at which the symbol would be displayed. The angle is positive (negative) if to the right (left) of the positive ORA as viewed by the pilot.
MC	!+ASL elevation+!	angle	Present angle between the ORA and the projection of the ray between the HUD origin and the symbol onto the plane formed by the ORA and the !!vertical ORA!! or (if the symbol is currently off) the elevation angle at which the symbol would be displayed. The angle is positive (negative) if to the above (below) of the positive ORA as viewed by the pilot.
C	!+ASL mode+!	HUD_ind_cntrl	Present display mode of the symbol.
C	!+ASL rotation+!	angle	The current rotation angle of the ASL, or (if the ASL is currently off) the rotation angle at which the ASL would be displayed. The angle is 0 degrees if the ASL is parallel to the Za axis.. The angle is 90 degrees if the ASL is parallel to Xa axis. 0 degrees <= !+ASL rotation+! < 360 degrees. The angle is measured clockwise from the !!vertical ORA!! as viewed from the pilot.
C	!+Aud signal+!	AUD_ind_cntrl	The current state of the audible signal.
M	!+Auto-cal sw+!	boolean	$true$ iff switch labeled “Auto-Cal” is set to the “on” position.
C	!+Auto-cal+!	vis_ind_cntrl	The state of the auto-cal indicator.
M	!+Az cursor lft max+!	angle	The leftmost (as seen by the pilot) position of the FLR azimuth cursor on the display screen; the minimum value of !+FLR az cursor posn+!.
M	!+Az cursor rgt max+!	angle	The rightmost (as seen by the pilot) position of the FLR azimuth cursor on the display screen; the maximum value of !+FLR az cursor posn+!.

TABLE 153. Variable Dictionary (Sheet 3 of 22)

MCT	Name	Type	Interpretation
M	!+az miss dist+!	distance	The distance along the ground between the target and the ground-projected line from the aircraft to the computed impact point.
M	!+az ref hdg pnl+!	angle	The last value of !+az ref hdg+! entered via the panel, or the default value if no value entered.
M	!+before slewing+!	boolean	true iff no legal slew inputs have yet been entered since entry into the current mode.
M	!+blast danger+!	boolean	true iff the pilot should immediately execute a 4g pullup to avoid dangerous weapon blast effects.
M	!+boc+!	boolean	$true$ iff the button labeled ‘‘BOC’’ is depressed.
M	!+bomb fall line+!	angle	!bomb fall line!. Imaginary line on the ground along which a ballistic weapon would travel if released now.
M	!+boresight azimuth+!	angle	The angle between the aircraft Ya axis and the weapon boresight in the Xa-Ya plane. The angle is negative if the weapon boresight points down when the aircraft is level.
M	!+boresight elevation+!	angle	The angle between the aircraft Ya axis and the weapon boresight in the Ya-ZA plane. The angle is negative if the weapon boresight line points to the left when viewed from above the level aircraft.
M	!+brg ac ftpt+!	angle	The angle measured clockwise (looking down) from the projection into the horizontal plane of the aircraft’s Ya axis to the projection into the horizontal plane of the line from the aircraft to the fly-to point. 0 <= !+brg ac ftpt+! < 360.
M	!+brg ac tgt+!	angle	The angle measured clockwise (looking down) from the projection into the horizontal plane of the aircraft’s Ya axis to the projection into the horizontal plane of the line from the aircraft to the target. 0 <= !+brg ac tgt+! < 360.
M	!+brg grtk cup+!	angle	The bearing measured clockwise (looking down) from the projection into the horizontal plane of the aircraft’s ground track to the projection into the horizontal plane of the line from the aircraft to the called-up point. 0 <= !+brg grtk cup+! < 360.
M	!+brg grtk ftpt+!	angle	The bearing measured clockwise (looking down) from the projection into the horizontal plane of the aircraft’s ground track to the projection into the horizontal plane of the line from the aircraft to the fly-to point. 0 <= !+brg grtk ftpt+! < 360.
M	!+brg grtk fxpt+!	angle	The bearing measured clockwise (looking down) from the projection into the horizontal plane of the aircraft’s ground track to the projection into the horizontal plane of the line from the aircraft to the fix point. 0 <= !+brg grtk fxpt+! < 360.
M	!+brg grtk oap+!	angle	The bearing measured clockwise (looking down) from the projection into the horizontal plane of the aircraft’s ground track to the projection into the horizontal plane of the line from the aircraft to the offset aim point. 0 <= !+brg grtk oap+! < 360.
M	!+brg grtk tgt+!	angle	The bearing measured clockwise (looking down) from the projection into the horizontal plane of the aircraft’s ground track to the projection into the horizontal plane of the line from the aircraft to the target. 0 <= !+brg grtk tgt+! < 360.

TABLE 153. Variable Dictionary (Sheet 4 of 22)

MCT	Name	Type	Interpretation
M	!+burst ht pnl+!	distance	
M	!+CA2 stage complete+!	boolean	true iff the named alignment mode stage has been completed since entering the current alignment mode. Note that this does not preclude the possibility of the stage being re-entered before the completion of the mode.
M	!+ccip+!	boolean	$true$ iff the button labeled ‘‘CCIP’’ is depressed.
M	!+central long a pnl+!	longitude	The last value of !+central long a+! entered via the panel, or the default value if no value entered.
C	!+central long a+!	longitude	The longitude of the central meridian of the A map; positive for east, negative for west.
M	!+central long b pnl+!	longitude	The last value of !+central long b+! entered via the panel, or the default value if no value entered.
C	!+central long b+!	longitude	The longitude of the central meridian of the B map; positive for east, negative for west.
M	!+CL stage complete+!	boolean	true iff the named test mode has been completed since entering the current test mode. The stage may or may not be re-entered before the test mode is complete.
M	!+CL2 stage complete+!	boolean	true iff the named test mode has been completed since entering the current test mode. The stage may or may not be re-entered before the test mode is complete.
C	!+Comp fail+!	boolean	$true$ iff the computer fail signal is on.
M	!+computed rls+!	boolean	true iff the active weapon would strike the target, or within an acceptable neighborhood of the target, if it were released right now.
M	!+CSA test result+!	boolean	true iff cycle-steal channel A and serial channel 1 pass built-in test.
M	!+CSB test result+!	boolean	true iff cycle-steal channel B and serial channel 2 pass built-in test.
M	!+cup ahead+!	boolean	true iff the called-up point is ahead of the aircraft; that is, iff the projection into the Xa-Ya plane of the line from the aircraft to the point has a positive Ya component.
M	!+data enterable+!	boolean	true iff a panel input operation may legally begin.
M	!+data nbr pnl+!	integer	The value entered by the pilot as part of the input procedure to discriminate among different panel input items. An integer from 0 to #max data nbr#, inclusively.
M	!+DC test result+!	boolean	true iff DC signal converter passes built-in test.
M	!+desig+!	boolean	true iff a reference point outside the a/c has been designated.
M	!+dest altitude pnl+!	distance	A value entered by the pilot that specifies the altitude of a destination.
M	!+dest entry pnl+!	integer	The value entered by the pilot to discriminate among different values of the same panel input item. An integer from #multval lbound# to #multval hbound#, inclusively.
M	!+dest lat+!	latitude	the latitude of destination !+loc nbr+!.
M	!+dest long+!	longitude	the longitude of destination !+loc nbr+!.
M	!+dest mslp pnl+!	pressure	The last value of !+dest mslp+! entered via the panel, or the default value if no value entered.

TABLE 153. Variable Dictionary (Sheet 5 of 22)

MCT	Name	Type	Interpretation
M	!+DIOW1 test result+!	boolean	true iff discrete input and output word pair 1 pass built-in test.
M	!+DIOW2 test result+!	boolean	true iff discrete input and output word pair 2 pass built-in test.
M	!+DIOW3 test result+!	boolean	true iff discrete input and output word pair 3 pass built-in test.
M	!+dive pullup+!	angle	!dive pullup!. If the A/C is close enought to the target, there exist two pitch angles (one greater than the pitch for maximum range, one less than this pitch) from which the aircraft, at present position and airspeed would release a !low drag! weapon, and the weapon would reach burst heifht at the target. If the aircraft is close emough for these two angles to be calculated, !dive pullup! is the lower of the two pitch angles minus the !system pitch!. Otherwise, !dive pullup! is 42 minus !system pitch!.
C	!+DME display+!	integer	All DME displays display the absolute value of !+DME display+!.
C	!+DME flag+!	boolean	The flag associated with the DME displays is displayed iff !+DME flag+! = $true$.
M	!+Doppler coupled pnl+!	boolean	The last value of !+Doppler coupled+! entered via the panel, or the default value if no value entered.
M	!+Doppler coupled+!	boolean	$true$ iff Doppler velocities should be used to damp system velocities and to calculate platform corrections when !+IMS mode+! = $Iner$.
M	!+Doppler reasonable+!	boolean	true iff the Doppler radar is producing reasonable groundspeed and drift angle readings.
M	!+Doppler up+!	boolean	true iff !+Doppler reasonable+!, and the groundspeed and drift angle readings produced by the Doppler are current.
M	!+drift angle IMS+!	angle	The drift angle of the aircraft, computed from IMS measurements, calculated by subtracting the !+heading IMS+! from the angle measured clockwise from Yp to the vector sum of !+N vel IMS+! and !+E vel IMS+!. -180 <= !+drift angle IMS+! < 180.
M	!+drift angle reliable+!	boolean	$true$ iff the DRS has not detected any internal malfunction which could render the measurement of drift angle invalid.
M	!+drift angle+!	angle	The horizontal angle between aircraft heading and the aircraft horizontal velocity vector, computed from the best available sources. The angle is positive when measured CW looking down from the heading to the velocity vectors. -180 <= !+drift angle+! <180.
M	!+drift test failed+!	boolean	true iff the current value of !+gyro drift delta n+! is too great.
M	!+drift test passed+!	boolean	true iff the current value of !+gyro drift delta n+! is small enough.
M	!+DRS mode+!	drs_mode	The current operating mode of the drs module.
C	!+DRS on+!	boolean	!+ DRS on+! = FALSE iff !+DRS mode+! = OFF.
M	!+during slewing+!	boolean	true iff a legal slewing operation is currently in progress.
MC	!+E coarse bias+!	accel	Measurement error for the Xp axis when the velocities are being measured by the coarse scale.
MC	!+E coarse scale+!	speed	Scale factor per pulse used for velocity calculation for the Xp axis when the velocities are being measured by the coarse scale.

TABLE 153. Variable Dictionary (Sheet 6 of 22)

MCT	Name	Type	Interpretation
MC	!+E fine bias+!	accel	Measurement error for the Xp axis when the velocities are being measured by the fine scale.
MC	!+E fine scale+!	speed	Scale factor per pulse used for velocity calculation for the Xp axis when the velocities are being measured by the fine scale.
M	!+E vel IMS+!	speed	Under certain conditions this term is the component of the aircraft velocity along the Xp axis as measured by the IMS and damped by a secondary source otherwise it is undefined. The value is positive in the positive Xp direction. The conditions are that the following sources on the Filter Behavior [FB] module interface are set: !!FB.speed 2!!, !!FB.speed 3!!, and !!FB.time inv 1!!. These sources are not set by DI.IMS.
M	!+elapsed navaln time+!	timeint	The elapsed time for which certain phases of alignment or navigation have proceeded, modulo the reset value, at which the measurement reverts to zero and resumes.
MC	!+Enter light+!	boolean	$true$ means that the associated indicator is turned on, $false$ means that it is turned off.
M	!+Enter pressed+!	boolean	$true$ iff the push button labeled “Enter” is pressed.
MC	!+failed state+!	boolean	no more software actions may occur more than #close down time# time after !+failed state+! becomes true.
M	!+flight path angle+!	angle	The angle from the aircraft velocity vector to its projection into the horizontal plane. It is positive (negative) when the aircraft velocity vector is positive (negative).
MC	!+FLR az cursor mode+!	FLR_ind_cntrl	The present display mode of the azimuth cursor.
C	!+FLR az cursor posn+!	angle	The azimuth currently being displayed by the FLR azimuth cursor, or (if the azimuth cursor mode is off) the position at which the cursor was last displayed.
C	!+FLR az+!	angle	The azimuth angle of the FLR pointing direction.
C	!+FLR elev+!	angle	The elevation angle of the FLR pointing direction.
MC	!+FLR mode+!	flr_mode	Current mode of the FLR. Initially $Idle$.
C	!+FLTDIR azimuth+!	angle	Present angle between the ORA and the projection of the ray between the HUD origin and the symbol onto the plane formed by the ORA and the !!horizontal ORA!! or (if the symbol is currently off) the azimuth angle at which the symbol would be displayed. The angle is positive (negative) if to the right (left) of the positive ORA as viewed by the pilot.
C	!+FLTDIR mode+!	HUD_ind_cntrl	Present display mode of the symbol.
M	!+Fly to num changed+!	boolean	True while !+Fly to num+! is changing value.
T	!+Fly to num reset+!	event	@C(!+Fly to num+!) WHEN (!+Fly to num+! != 0 & PREV(!+Fly to num+!) != 0)
M	!+Fly to num+!	integer	The setting of the numeric selector labeled “Fly to”.

TABLE 153. Variable Dictionary (Sheet 7 of 22)

MCT	Name	Type	Interpretation
M	!+Fly to state changed+!	boolean	True while !+Fly to state+! is changing value.
M	!+Fly to state+!	fly_to_state	The setting of the selector labeled “Fly to”.
M	!+FM stage complete+!	boolean	true iff the FM alignment mode stage has been completed since entering the current alignment mode. Note that his does not preclude the possibility of the stage being re-entered before the completion of the mode.
C	!+Format U321+!	boolean	$true$ iff display format 321 for the upper window is on.
MC	!+FPM azimuth+!	angle	Present angle between the ORA and the projection of the ray between the HUD origin and the symbol onto the plane formed by the ORA and the !!horizontal ORA!! or (if the symbol is currently off) the azimuth angle at which the symbol would be displayed. The angle is positive (negative) if to the right (left) of the positive ORA as viewed by the pilot.
MC	!+FPM elevation+!	angle	Present angle between the ORA and the projection of the ray between the HUD origin and the symbol onto the plane formed by the ORA and the !!vertical ORA!! or (if the symbol is currently off) the elevation angle at which the symbol would be displayed. The angle is positive (negative) if above (below) of the positive ORA as viewed by the pilot.
C	!+FPM mode+!	HUD_ind_cntrl	Present display mode of the symbol.
M	!+ftpt ahead+!	boolean	true if and only if the fly-to point is ahead of the aircraft; that is, iff the projection into the Xa-Ya plane of the line from the aircraft to the point has a positive Ya component.
M	!+fxpt ahead+!	boolean	true if and only if the fix point is ahead of the aircraft; that is, iff the projection into the Xa-Ya plane of the line from the aircraft to the point has a positive Ya component.
M	!+GAS+!	boolean	true iff the current steering state is go-around-steering.
M	!+gnd speed DRS+!	speed	The magnitude of the projection of the A/C velocity vector onto the horizontal plane. When the mode is OPERATE this is the current measurement. When the mode is MEMORY this is a stale value.
M	!+gnd speed reliable+!	boolean	$true$ iff the DRS has not detected any internal malfunction which could render the measurement of groundspeed invalid.
M	!+gr ac ftpt+!	distance	The ground range from the aircraft’s present position to the fly-to point.
M	!+gr ac fxpt+!	distance	The ground range from the aircraft’s present position to the fix point.
M	!+gr ac HUDrefpt+!	distance	The ground range from the aircraft’s present position to the HUD reference point.
M	!+gr ac oap+!	distance	The ground range from the aircraft’s present position to the offset aim point.
M	!+gr ac rmax+!	distance	The ground range between the aircraft’s present position and the position where the condition !+rmax+! will be true.
M	!+gr ac stik exit+!	distance	The ground range from the aircraft’s present position to the point at which @T(!+stik empty+!) last occurred.
M	!+gr ac tgt+!	distance	The ground range from the aircraft’s present position to the target.

TABLE 153. Variable Dictionary (Sheet 8 of 22)

MCT	Name	Type	Interpretation
M	!+ground danger+!	boolean	true iff the pilot must execute an immediate 4g pullup to avoid striking the ground.
T	!+ground tests finished+!	boolean	@F(!+in Grtest+!)
M	!+grtk+!	angle	The bearing measured clockwise (looking down) from (a) the projection into the horizontal plane of a line from the a/c to true North, to (b) the projection into the horizontal plane of the aircraft's velocity vector.
M	!+Gun Enable+!	boolean	$true$ iff the gun is currently enabled
T	!+Guns+!	boolean	!+Weapon Class+! = GN
M	!+gyro drift delta n+!	angrate	The difference between the latest value of !+Y drift+! and (1) the previous value, or (2) 0 deg/hour if there is no previous value. The value is updated in $01update$ mode during TS alignment stage. Also, the value is set to 0 deg/hour when @T(!+in $landaln$) occurs.
M	!+hdg system+!	angle	a/c heading. 0 <= !+hdg system+! <360.
M	!+heading IMS+!	angle	If !+IMS rel+! is $true$ then the angle measured clockwise in the Xp-Yp plane (looking in the positive Zp direction) from the Yp axis to the projection in the Xp-Yp plane of the Ya axis; otherwise it is undefined. 0 <= !+heading IMS+! < 360.
M	!+heading MAG+!	angle	If !+IMS rel+! is $true$ then the magnetic heading of the aircraft; otherwise it is undefined. Magnetic heading is the angle measured CW from magnetic north (looking down) to the horizontal component of the Ya axis. This value is not affected by the alignment of the platform. 0 <= !+heading MAG+! < 360.
M	!+high drag release+!	boolean	true iff a weapon type with alterable delivery characteristics has been chosen, and the pilot has selected the high drag configuration.
M	!+High Drag+!	boolean	$true$ iff the switch labeled "RET WPN" is selected.
M	!+HS stage complete+!	boolean	true iff the named alignment mode stage has been completed since entering the current alignment mode. Note that this does not preclude the possibility of the stage being re-entered before the completion of the mode.
C	!+HSI 1+!	angle	HSI pointer 1 indicate angle !+HSI 1+! measured CW from the reference point as seen by the pilot.
C	!+HSI 2+!	angle	HSI pointer 2 indicate angle !+HSI 2+! measured CW from the reference point as seen by the pilot.
C	!+HUD alt+!	distance	The current value being shown on the HUD altitude display.
C	!+HUD heading+!	angle	The current value being shown on the HUD heading display.
MC	!+HUD NACC+!	accel	The current value being shown on the HUD normal acceleration display, or (if the display is currently off or not under software control) the value that would be shown on the HUD normal acceleratioin display.
C	!+HUD pitch+!	angle	The current value being shown on the HUD pitch display.

TABLE 153. Variable Dictionary (Sheet 9 of 22)

MCT	Name	Type	Interpretation
M	!+HUD reliable+!	boolean	$true$ iff HUD is ready and has passed its most recent built-in test. $false$ if HUD is either not ready or has failed its built-in test. This state does not ensure that symbols are not being displayed; only that the HUD *may* have failed.
C	!+HUD roll+!	angle	The current value being shown on the HUD roll display.
C	!+HUD test mode+!	HUD_test	A indicats that test pattern A is being displayed. B indicates that test patterrn B is being displayed. $None$ indicates that no test pattern is being displayed.
C	!+HUD vertvel+!	speed	The current value being shown on the HUD vertical velocity display, or (if the display is currently off or not under software control) the value that would be shown on the HUD vertical velocity display.
M	!+HUDrefpt az+!	angle	The angle between the Ya axis, and the projection into the Xa-Ya plane of the ray from the aircraft to the current HUD reference point. The angle is positive if the ray is to the right (looking down) of the Ya axis.
M	!+HUDrefpt elev+!	angle	The angle between the Ya axis, and the projection into the Ya-Za plane of the ray from the aircraft to the current HUD reference point. The angle is positive if the ray is above (positive Za direction) the plane.
M	!+IMS coarse rotating+!	boolean	$true$ iff coarse rotation in progress. Initialized to $false$ at system generation time.
C	!+IMS E velocity+!	speed	Initialize E IMS velocity to this value.
C	!+IMS enable+!	boolean	@T(!+IMS enable+!) enables the IMS device for computer control. @F(!+IMS enable+!) disables IMS device for computer control and the coarse rotation in progress (if any) is stopped.
M	!+IMS horiz velocity+!	speed	Under certain conditions this term is the component of the aircraft velocity in the Xp-Yp plane as measured by the IMS and damped by a secondary source; otherwise it is undefined.
M	!+IMS mode+!	imsmode	The current setting of the IMS mode switch. When the mode is Off, attitude and velocity measurements are meaningless, and software actions are not permitted.
C	!+IMS N velocity+!	speed	Initialize N IMS velocity to this value.
M	!+IMS ready+!	boolean	$true$ iff IMS is ready for operation under computer control.
M	!+IMS reasonable+!	boolean	true iff the IMS is giving reasonable results.
M	!+IMS rel+!	boolean	$true$ iff the IMS is to be considered reliable based on its internal self-test.
C	!+IMS scale+!	imsscale	When !+IMS scale+! = $Fine$ use the finer scale for north and east velocity measurements. When !+IMS scale+! = $Coarse$ use the coarser scale for north and east velocity measurements.
M	!+IMS total velocity+!	speed	Under certain conditions this term is the aircraft velocity as measured by the IMS and damped by a secondary source; otherwise it is undefined.
M	!+IMS up+!	boolean	true iff the IMS has completed its built-in self-alignment and has passed its most recent built-in self-test.
C	!+IMS V velocity+!	speed	Initialize V IMS velocity to this value.
M	!+in flight+!	boolean	true iff the a/c is airborne.

TABLE 153. Variable Dictionary (Sheet 10 of 22)

MCT	Name	Type	Interpretation
M	!+in *x*+!	boolean	(where *x* is replaced by a mode name, without the $ brackets and without any prefix; for example: !+in airaln+!) True iff the system is in the mode denoted by *x*.
M	!+Init complete+!		Set to false when @T(!+Power up+!) occurs. Set to true when all modules in the system have completed their power-up initialization procedures.
M	!+input attempted+!	boolean	true while the pilot is attempting to enter input through the panel without issuing a preliminary keyboard command to begin the operation.
M	!+input requested+!	boolean	true while the pilot is issuing the preliminary keyboard command to begin an input operation.
M	!+interrupt test result+!	boolean	true iff the interrupt hardware passes built-in test.
M	!+ip az+!	angle	impact point azimuth
M	!+ip elev+!	angle	impact point elevation
M	!+Keybd input+!	keybd	Identifies the data button pressed since the last time +G_KEYBD_INPUT+ was called. Equals $Error$ if more than one data button has been pressed since the last time +G_KEYBD_INPUT+ was called. Equals $None$ if no data button has been pressed since the last call.
M	!+L-probe pnl+!	boolean	The last value of !+L-probe+! entered via the panel, or the default value if no value entered.
MC	!+L-probe+!	boolean	$true$ indicates the ADC module is currently operating as it should for a device with an L-probe; $false$ means it is operating for a device with a non-L-probe.
M	!+land based pnl+!	boolean	$true$ indicates the aircraft is land based. $false$ indicates the aircraft is sea (carrier) based.
M	!+land velocity test failed+!	boolean	true iff the land velocity test is not considered to have passed. The land velocity test is a test performed to determine the reliability of the IMS velocity measurements while the a/c is not airborne.
M	!+land velocity test passed+!	boolean	true iff the land velocity test is considered to have been passed. The land velocity test is a test performed to determine the reliability of the IMS velocity measurements while the a/c is not airborne.
M	!+latitude cup+!	latitude	The latitude of the called-up point.
M	!+latitude error+!	latitude	The difference in latitude between the two current positional reference points.
M	!+latitude+!	latitude	The first element of the current !+a/c location+!.
M	!+loc nbr+!	integer	A number indicating a point on the earth.
M	!+loft pullup+!	angle	!loft pullup!. The difference between the upper release angle for !low drag! weapons defined in !dive pullup! and the !system pitch!. If the aircraft is close enough for these two angles to be calculated, !loft pullup! is the upprt of the two pitch angles minus teh !system pitch!. Otherwise, !loft pullup! is 42 minus the !system pitch!.
M	!+longitude cup+!	longitude	The longitude of the called-up point.
M	!+longitude error+!	longitude	The difference in longitude between the two current positional reference points.

TABLE 153. Variable Dictionary (Sheet 11 of 22)

MCT	Name	Type	Interpretation
M	!+longitude+!	longitude	The second element of the current !+a/c location+!.
M	!+low drag release+!	boolean	true iff a weapon type with alterable delivery characteristics has been chosen, and the pilot has selected the low drag configuration.
M	!+low lat ct a pnl+!	integer	The last value of !+low lat ct a+! entered via the panel, or the default value if no value entered.
C	!+low lat ct a+!	integer	A signed number representing the southern most latitude of the A map area covered; this number is in counts, where each count represents 8/9 of a degree from the equator. The number is positive for a north latitude; negative for a south latitude.
M	!+low lat ct b pnl+!	integer	The last value of !+low lat ct b+! entered via the panel, or the default value if no value entered.
C	!+low lat ct b+!	integer	A signed number representing the southern most latitude of the B map area covered; this number is in counts, where each count represents 8/9 of a degree from the equator. The number is positive for a north latitude; negative for a south latitude.
C	!+low win fmt+!	format	Formatting of the lower window display.
C	!+low win val+!	char_string_7	Display !+low win val+! in the lower window.
M	!+LSC az on ASL+!	angle	If !+ASL rotation+! = 90 or 270 then !+ALS rotation+!. Otherwise, the azimuth angle of a point on the HUD azimuth steering line (ASL) symbol at elevation equal to !+LSC elevation+!. The ASL symbol is taken to be a line segment arbitrarily long; if no point on the actual symbol is at the given elevation, the result will be calculated as though the segment were long enough to reach that elevation. The resolution of !+LSC elevation+! is assumed not to be less than #HUD symbol el res#. The computed result is limited to be within the HUD field of view; #HUD symbol az max# >= !+LSC az on ASL+! >= #HUD symbol az min#).
MC	!+LSC azimuth+!	angle	Present angle between the ORA and the projection of the ray between the HUD origin and the symbol onto the plane formed by the ORA and the !!horizontal ORA!! or (if the symbol is currently off) the azimuth angle at which the symbol would be displayed. The angle is positive (negative) if to the right (left) of the positive ORA as viewed by the pilot.
MC	!+LSC elevation+!	angle	Present angle between the ORA and the projection of the ray between the HUD origin and the symbol onto the plane formed by the ORA and the !!vertical ORA!! or (if the symbol is currently off) the elevation angle at which the symbol would be displayed. The angle is positive (negative) if above (below) of the positive ORA as viewed by the pilot.
MC	!+LSC mode+!	HUD_ind_cntrl	Present display mode of the symbol.
M	!+mag variation pnl+!	angle	The last value of !+mag variation+! entered via the panel, or the default value if no value entered.
M	!+Map decenter+!	boolean	$true$ iff switch labeled “Decenter” on PMDS set to on position.
M	!+Map displayable+!	boolean	$true$ iff the location !!refpt lat!! and !!refpt long!! can be displayed at the current scale.

TABLE 153. Variable Dictionary (Sheet 12 of 22)

MCT	Name	Type	Interpretation
M	!+Map hold changed+!	boolean	True while !+Map hold+! is changing value.
M	!+Map hold+!	boolean	$true$ iff switch labeled “Hold” on PMDS set to on.
C	!+Map indicator+!	angle	The indicator on the map screen is set to indicate the value of !+Map Indicator+!.
MC	!+Map latitude+!	latitude	The geographic latitude under the map reference point on the screen.
M	!+Map ldg+!	boolean	$true$ iff switch labeled “Map landing” on PMDS set to on position.
MC	!+Map longitude+!	longitude	The geographic longitude under the map reference point on the screen.
M	!+Map north-up+!	boolean	$true$ iff switch labeled “North-Up” on PMDS set to on position.
M	!+map orient a pnl+!	angle	The last value of !+map orient a+! entered via the panel, or the default value if no value entered.
C	!+map orient a+!	angle	The angle between a meridian line on the A map and the longitudinal axis of the map. Map orientation is zero degrees for north/south maps, 90 degrees for east/west maps.
M	!+map orient b pnl+!	angle	The last value of !+map orient b+! entered via the panel, or the default value if no value entered.
C	!+map orient b+!	angle	The angle between a meridian line on the B map and the longitudinal axis of the map. Map orientation is zero degrees for north/south maps, 90 degrees for east/west maps.
C	!+Map pointer angle+!	angle	The current angle of the software controlled pointer on the map screen. The angle is measured clockwise from the top center of the display. 0 <= !+Map pointer angle+! < 360.
C	!+Map position valid+!	boolean	$true$ if a valid map display is under the reference point. $false$ indicates that the map warning is visible under the reference point.
C	!+Map ref pt+!	ref_pt	The map reference point is the point on the map display screen under which the requested location on the map is positioned. $center$ indicates the reference point at the center of the screen. $bottom-center$ indicates the reference point at the bottom center edge of the screen.
C	!+Map rotation+!	angle	The angle from (a) the line from the center of the display to north on the map to (b) the line from the center of the display to the top-center of the display has the value !+Map rotation+!. 360 > !+Map rotation+! >= 0.
M	!+Map scale sw+!	boolean	$true$ iff toggle set to “80”.
C	!+Map scale+!	map_scale	The scale at which the map is displayed.
C	!+Map warning+!	boolean	$true$ iff the map screen shows a distinctive display, such as all hashmarks.
M	!+mark lat+!	latitude	The latitude of mark position !+mark loc nbr+!.
M	!+mark loc nbr+!	integer	A number indicating a point on the earth.
M	!+mark long+!	longitude	The longitude of mark position !+mark loc nbr+!.
M	!+Mark pressed+!	boolean	$true$ iff the push button labeled “Mark” is pressed.
C	!+Mark window+!	char	The alphanumeric character displayed in the Mark window.

TABLE 153. Variable Dictionary (Sheet 13 of 22)

MCT	Name	Type	Interpretation
M	!+mark+!	integer	The number associated with the most recently defined Mark destination.
M	!+Master Arm+!	boolean	$true$ iff the master arm switch is selected.
M	!+memory test result+!	boolean	true iff the memory diagnostic is passed.
M	!+Mult Rack+!	boolean	$true$ iff the active weapon station contains a multiple or triple ejector rack. Valid only if at least one element of !+Ready Stations+! = $true$.
MC	!+N coarse bias+!	accel	Measurement error for the Yp axis when the velocities are being measured by the coarse scale.
MC	!+N coarse scale+!	speed	Scale factor per pulse used for velocity calculation for the Yp axis when the velocities are being measured by the coarse scale.
MC	!+N fine bias+!	accel	Measurement error for the Yp axis when the velocities are being measured by the fine scale.
MC	!+N fine scale+!	speed	Scale factor per pulse used for velocity calculation for the Yp axis when the velocities are being measured by the fine scale.
C	!+N light+!	boolean	$true$ means that the "N" indicator is turned on, $false$ means that it is turned off.
M	!+N vel IMS+!	speed	Under certain conditions this term is the component of the aircraft velocity along the Yp axis as measured by the IMS and damped by a secondary source; otherwise it is undefined. The value is positive in the positive Yp direction. The conditions are that the following sources on the Filter Behavior [FB] module are set: !!FB.speed 5!!, !!FB.speed 6!!, and !!FB.time inv 2!!. These sources are not set by DI.IMS.
M	!+natt+!	boolean	$true$ iff the button labeled "NORM ATTACK" is depressed.
M	!+nav velocity test failed+!	boolean	true iff the differences between the Doppler- and IMS-measured velocities are not within acceptable bounds when the system is in a navigation mode in which the Doppler provides a backup reference velocity. Note that this is not the opposite of !+Air velocity test passed+!, as the acceptable bounds may differ in the two cases.
M	!+nbr rdy sta+!	integer	The number of weapon stations that are currently ready; $0 \leq$!+nbr rdy sta+! $\leq$ #num weap stations#.
M	!+ND2 stage complete+!	boolean	true iff the named alignment mode stage has been completed since entering the current alignment mode. Note that this does not preclude the possibility of the stage being re-entered before the completion of the mode.
M	!+new align stage+!	boolean	becomes true each time the alignment stage changes. This includes an entry into an alignment stage from no alignment stage. This does not include entering no alignment stage.
M	!+new *data* entered+!	boolean	Signalled when the pilot has just entered a fresh value (whether it is equal to the current value or not) for !+*data* pnl+!, which is defined for each *data* elsewhere in this dictionary.
M	!+new dest coords entered+!	boolean	Signalled when the pilot has just entered new values for !+dest lat pnl+! and !+dest long pnl+! in a single input operation.
M	!+new posn entered+!	boolean	Signalled when the pilot has just entered new values for !+latitude pnl+! and !+longitude pnl+! in a single input operation.

TABLE 153. Variable Dictionary (Sheet 14 of 22)

MCT	Name	Type	Interpretation
M	!+No intervening take-off+!	boolean	@T(!+in flight+!) has not occurred since last time !+in Landaln+! became false.
MC	!+Non-align+!	VIS_ind_cntrl	The state of the non-align indicator.
M	!+normal accel+!	accel	The acceleration of the A/c along the aircraft Za axis; positive in the Za positive direction.
M	!+oap ahead+!	boolean	true if and only if the offset aim point is ahead of the aircraft; that is, iff the projection into the Xa-Ya plane of the line from the aircraft to the point has a positive Ya component.
M	!+offset brg pnl+!	angle	The last value of !+offset brg+! entered via the panel, or the default value if no value entered.
M	!+offset dht pnl+!	distance	A value entered by the pilot that specifies the difference in altitude between an offset aim point and a target.
M	!+offset rng pnl+!	distance	The last value of !+offset rng+! entered via the panel, or the default value if no value entered.
M	!+offset+!	boolean	$true$ iff the button labeled ‘‘OFFSET’’ is depressed.
T	!+Other Weapon+!	boolean	!+Station selected+! & !+Weapon Class+! != UN & !+Weapon Class+! != GN & NOT (!+Reserved Weapon+! Or !+Shrike+!)
M	!+OTS pullup+!	angle	!OTS pullup!. With !low drag! weapons and !OTS!, the A/C must pullup to reach the correct release angle in an OTS maneuver. In the first portion of the pullup !A/C facing target! is not true. During this time interval the !loft pullup! is calculated for a !low drag! weapon to an imaginary target that is the same !ground range! as !+gr ac tgt+! but at a bearing from the aircraft that is 180 degrees from the bearing of the target. the !OTS pullup! is 180 minus (2 X !system pitch!). When !A/C facing target! and !OTS!, !OTS pullup! is equal to !loft pullup!.
M	!+OTS+!	boolean	true iff the current steering state is over-the-shoulder steering.
M	!+Overflown exit+!	boolean	True if last weapon mode was exited because the target was overflown without a release.
M	!+panel error+!	boolean	true while the panel is displaying the error message.
M	!+Panel mode changed+!	boolean	True while !+Panel mode+! is changing value.
M	!+Panel mode+!	panel_mode	The setting of the panel mode selector switch. Values correspond to switch nomenclature.
M	!+pitch IMS+!	angle	If !+IMS rel+! is $true$ then the angle between the Ya axis and its projection into the Xp-Yp plane; otherwise it is undefined. The angle is negative when the positive Ya axis is below the Xp-Yp plane, and positive otherwise. -90 <= !+pitch IMS+! < 90.
T	!+pitch small+!	boolean	ABS (!+pitch system+!) <= 20
M	!+pitch system+!	angle	a/c pitch. -90 <= !+pitch system+! <90.
M	!+pnl config changed+!	boolean	True while !+pnl config+! is changing from one set of values to another.

TABLE 153. Variable Dictionary (Sheet 15 of 22)

MCT	Name	Type	Interpretation
M	!+pnl config+!	panel_config	A current state of the panel control configuration. !+pnl config+! may have more than one value at a time, provided that the current set of values does not include $none$.
M	!+power up+!	boolean	computer is in the operating state and may be assumed to be functioning properly.
M	!+preparation time+!	timeint	The warmup time required by this weapon before release.
C	!+prepare weapon+!	boolean	On occurrence of the event @C(!+prepare weapon+!) preparations (required by certain weapons) commence for the weapon(s) on the currently active station(s). If the !+preparation time+! of the weapon(s) is zero or undefined, changes to this variable have no effect.
M	!+Pres pos changed+!	boolean	True while !+Pres pos+! is changing value.
M	!+Pres pos+!	pp_mode	The setting of the present position selector switch. Values correspond to switch nomemclature.
M	!+PUAC az on ASL+!	angle	If !+ASL rotation+! = 90 or 270 then !+ALS rotation+!. Otherwise, the azimuth angle of a point on the HUD azimuth steering line (ASL) symbol at elevation equal to !+PUAC elevation+!. The ASL symbol is taken to be a line segment arbitrarily long; if no point on the actual symbol is at the given elevation, the result will be calculated as though the segment were long enough to reach that elevation. The resolution of !+PUAC elevation+! is assumed not to be less than #HUD symbol el res#. The computed result is limited to be within the HUD field of view; #HUD symbol az max# >= !+PUAC az on ASL+! >= #HUD symbol az min#).
C	!+PUAC azimuth+!	angle	Present angle between the ORA and the projection of the ray between the HUD origin and the symbol onto the plane formed by the ORA and the !!horizontal ORA!! or (if the symbol is currently off) the azimuth angle at which the symbol would be displayed. The angle is positive (negative) if to the right (left) of the positive ORA as viewed by the pilot.
MC	!+PUAC elevation+!	angle	Present angle between the ORA and the projection of the ray between the HUD origin and the symbol onto the plane formed by the ORA and the !!vertical ORA!! or (if the symbol is currently off) the elevation angle at which the symbol would be displayed. The angle is positive (negative) if above (below) of the positive ORA as viewed by the pilot.
C	!+PUAC mode+!	HUD_ind_cntrl	Present display mode of the symbol.
C	!+PUC mode+!	HUD_ind_cntrl	Present display mode of the symbol.
M	!+r65+!	boolean	true iff the a/c is in the proper configuration to release a special weapon at a 65 degree flight path angle.
M	!+radalt priority pnl+!	boolean	The last value of !+radalt priority+! entered via the panel, or the default value if no value entered.
M	!+RE pressed+!	boolean	$true$ iff the release enable button is currently pressed.
M	!+Ready Stations+!	boolean array	This array is indexed by the weapon station numbers (from 1 to #num weap stations#). A value is $true$ iff the corresponding station is a *ready* station. Only ready stations are eligible to be active stations.

TABLE 153. Variable Dictionary (Sheet 16 of 22)

MCT	Name	Type	Interpretation
T	!+Redesignate+!	boolean	(!+TD pressed+! OR (!+keybd input+! != 0 & !+keybd input+! != $None$)) & !+in mode AflyUpd+!
M	!+Rel in Progress+!	boolean	$true$ iff a release pulse is currently being issued to the active weapon station(s).
M	!+release pulse width+!	timeint	The release pulse width required by this weapon.
C	!+release weapon+!	boolean	On occurrence of the event @C(!+release weapon+!) the fire ready and bomb release signals are sent to the active weapon stations(s) for !+release pulse width+! length of time.
T	!+Reserved Weapon+!	boolean	!+Walleye+! OR !+Special+! OR !+Rockets+! OR !+Guns+!
M	!+rls pts passed+!	integer	The number of computed release points for the latest stik that have already been passed. Value upon entry to a weapon mode is undefined; set to 0 when @T(!+stik created+!) occurs.
M	!+rmax+!	boolean	true iff the a/c is in the proper configuration and at a maximum !!pullup range!! that would result in a special weapon impacting the target.
M	!+rmax+6000+!	boolean	true iff the a/c is in the proper configuration and that the a/c is a !!pullup range!! that would result in a special weapon impacting 6000 feet short of the target.
M	!+rmin+!	boolean	true iff the a/c is in the proper configuration and that the a/c is at a minimum !!pullup range!! that would result in a special weapon impacting the target.
M	!+rmin+6000+!	boolean	true iff the a/c is in the proper configuration and that the a/c is at a !!pullup range!! that would result in a special weapon impacting 6000 feet long of the target.
C	!+Rng cursor+!	distance	The range currently being displayed by the FLR cursors.
C	!+RNGCUE mode+!	HUD_ind_cntrl	Present display mode of the symbol.
T	!+Rockets+!	boolean	!+Weapon Class+! = RK
M	!+roll IMS+!	angle	If !+IMS rel+! is $true$ then the angle between the Xa axis and the direction defined by the cross-product of the positive Ya axis and the positive Zp axis; otherwise it is undefined. The angle is positive when the positive Xa axis is below the Xp-Yp plane (positive Zp component) and negative otherwise. -180 <= !+roll IMS+! < 180.
T	!+roll small+!	boolean	ABS (!+roll system+!) <= 30
M	!+roll system+!	angle	a/c roll. -180 <= !+roll system+! < 180.
C	!+sea level pressure+!	pressure	Atmospheric pressure at sea level.
M	!+Self-test+!!	boolean	$true$ iff switch labeled “Test” on panel set to on position.
C	!+set release pulse width+!	timeint	Length of time to issue fire ready and bomb release signals to the active weapon stations.
T	!+Shrike+!	boolean	!+Weapon Class+! = SK
M	!+SINS dhdg pnl+!	angle	The last value of !+SINS dhdg+! entered via the panel, or the default value if no value entered.
M	!+SINS east vel valid+!	boolean	$true$ iff SINS east velocity data is valid.

TABLE 153. Variable Dictionary (Sheet 17 of 22)

MCT	Name	Type	Interpretation
M	!+SINS east vel+!	speed	The Xs component of the ship's velocity. The value is positive in the positive Xs direction.
C	!+SINS enabled+!	boolean	The SINS device interface is enabled iff !+SINS enabled+! = true.
M	!+SINS heading valid+!	boolean	$true$ iff SINS heading data is valid.
M	!+SINS heading+!	angle	The angle measured clockwise from the line from the earth's north to the ship and the Ys axis looking down onto the ship from above. 0 <= !+SINS heading+! < 360.
M	!+SINS lat valid+!	boolean	$true$ iff SINS latitude data is valid.
M	!+SINS lat+!	latitude	The latitude of the ship as indicated by the SINS data.
M	!+SINS long valid+!	boolean	$true$ iff SINS longitude data is valid.
M	!+SINS long+!	longitude	The longitude of the ship as indicated by the SINS data.
M	!+SINS north vel valid+!	boolean	$true$ iff SINS north velocity data is valid.
M	!+SINS north vel+!	speed	The Ys component of the ship's velocity as indicated by the SINS data. The value is positive in the positive Ys direction.
M	!+SINS pitch valid+!	boolean	$true$ iff SINS pitch data is valid.
M	!+SINS roll valid+!	boolean	$true$ iff SINS roll data is valid.
M	!+SINS velocity test passed+!	boolean	true iff the IMS-measured velocities are close enough to the SINS-measured velocities when compared.
M	!+SINS x offset pnl+!	distance	The last value of !+SINS x offset+! entered via the panel, or the default value if no value entered.
M	!+SINS y offset pnl+!	distance	The last value of !+SINS y offset+! entered via the panel, or the default value if no value entered.
M	!+SINS z offset pnl+!	distance	The last value of !+SINS z offset+! entered via the panel, or the default value if no value entered.
T	!+Slew displacement non-zero+!	boolean	$true$ iff !+Slew right-left+! != 0 OR !+Slew up-down+! != 0.
M	!+slew FLR delta az+!	angle	How much a slewed FLR symbol should be shifted in azimuth from its current position, given the current slew control position. Positive value means right (as seen by the pilot).
M	!+slew FLR delta rng+!	distance	How much a slewed FLR symbol should be shifted in range from its current position, given the current slew control position. Positive value means range location of symbol should be increased.
M	!+slew HUD delta az+!	angle	How much a slewed HUD symbol should be shifted in azimuth from its current position, given the current slew control position. Positive value means right (as seen by the pilot).
M	!+slew HUD delta elev+!	angle	How much a slewed HUD symbol should be shifted in elevation from its current position, given the current slew control position. Positive value means up (as seen by the pilot).
M	!+slew map delta lat+!	latitude	How much the map display should be shifted in latitude from its current position, given the current slew control position.

TABLE 153. Variable Dictionary (Sheet 18 of 22)

MCT	Name	Type	Interpretation
M	!+slew map delta long+!	longitude	How much the map display should be shifted in longitude from its current position, given the current slew control position.
M	!+Slew right-left+!	real	The right-left component of the current slew displacement. If the control is in the center position this value is zero. A positive value indicates that the control is right of center and a negative value indicates left of center. The extreme values of the range indicate that the control is at its limit of right-left movement. The value is directly proportional to the amount of displacement in the left-right axis. In this context, "left" refers to the negative Xa direction, and "right" to the positive Xa direction.
M	!+Slew up-down+!	real	The up-down component of the current slew displacement. If the control is in its center position this value is zero. A positive value indicates that the control is above center (up) and a negative value indicates below center (down). The extreme values of the range indicate that the control is at its limit of up-down movement. The value returned is directly proportional to the amount of displacement in the up-down axis. In this context, "up" refers to the positive Ya direction, and "down" to the negative Ya direction.
M	!+sm drftang DRS+!	angle	Smoothed value of !+drift angle DRS+!.
M	!+sm gndspd DRS+!	speed	Smoothed value of !+gnd speed DRS+!.
M	!+special in range+!	boolean	true iff the target is in range for a special weapon.
T	!+Special+!	boolean	!+Weapon Class+! = SOD OR !+Weapon Class+! = SSH
M	!+sr ac btpup+!	distance	The slant range (straight-line) distance to the point where @T(!+blast danger+!) will occur due to weapon blast effects should the a/c continue its present course.
M	!+sr ac gpup+!	distance	The slant range (straight-line) distance to the point where @T(!+ground danger+!) will occur due to ground proximity should the a/c continue its present course.
M	!+sr ac ip+!	distance	The slant range from the a/c's present position to the computed impact point of the next weapon in the stik.
M	!+sr ac rls+!	distance	The slant range (straight-line distance) from the a/c to the point where the next weapon release should occur.
M	!+sr ac tgt+!	distance	The slant range from the aircraft to the target.
M	!+sr reasonable+!	boolean	true iff the a/c attitude and FLR configuration are such that reliable FLR slant range readings may be taken, and the FLR slant range value is reasonable.
M	!+steering error to rls+!	angle	The angle between the a/c ground track and the horizontal line from the a/c to the point where the a/c should release the current weapon in order to strike the target. Positive (negative) if the ground track line is to the left (right) of the line to the release point, looking down. -180 <= !+steering error to rls+! < +180.
M	!+steering error to tgt+!	angle	The angle between the a/c ground track and the horizontal line from the a/c to the target. Positive (negative) if the ground track line is to the left (right) of the line to the target, looking down. -180 <= !+steering error to tgt+! < +180.
M	!+steering to tgt+!	boolean	true iff the current steering state is steering-to-target.

TABLE 153. Variable Dictionary (Sheet 19 of 22)

MCT	Name	Type	Interpretation
M	!+stik created+!	boolean	Initialized to false at weapon mode entry. Becomes true when the first release point in a stik is identified but the aircraft has not yet reached it. Becomes false when @T(!+stik empty+!) occurs.
M	!+stik empty+!	boolean	Initialized to true at weapon mode entry. Becomes false when @T(!+stik created+!) occurs. Becoms true when the aircraft has passed the last release point in the current stik.
M	!+target in range+!	boolean	true iff the target is in range of the current weapon type.
M	!+TAS ADC valid+!	boolean	If !+TAS ADC valid+! then !+TAS ADC+! is the unfiltered true airspeed of the aircraft as measured by the ADC. Otherwise, !+TAS ADC+! is undefined.
M	!+TAS ADC+!	speed	If !+TAS ADC valid+! then !+TAS ADC+! is the unfiltered true airspeed of the aircraft as measured by the ADC. Otherwise, !+TAS ADC+! is undefined.
M	!+TD pressed+!	boolean	$true$ iff the target designate button is pressed.
M	!+test stage+!	tstage	The current test mode stage of the system. Value is $None$ if the system is in no test stage.
M	!+tf+!	boolean	$true$ iff the button labeled ‘‘TF’’ is depressed.
M	!+tgt ahead+!	boolean	true if and only if the target is ahead of the aircraft; that is, iff the projection into the Xa-Ya plane of the line from the aircraft to the point has a positive Ya component.
M	!+time to ftpt+!	timeint	The time to go before the a/c reaches the fly-to point, assuming a direct flight toward the point at the current horizontal velocity. Always positive.
T	!+time to prepare+!	boolean	true iff !+time to rls+! <= !+preparation time+! for the current weapon. This term is never true for those weapons for which !+preparation time+! is not defined.
M	!+timer test result+!	boolean	true iff the timer hardware passes built-in test.
M	!+TOS+!	boolean	true iff the current steering state is tail-on-steering.
M	!+TS stage complete+!	boolean	True iff the TS alignment mode state has been completed since entering the current alighnment mode. Note that this does not preclude the possibility of the stage being re-entered before the completion of the mode.
C	!+up win fmt+!	format	Formatting of the upper window display.
C	!+up win val+!	char_string_6	Display !+up win val+! in the upper window.
M	!+Update changed+!	boolean	True while !+Update+! is changing value.
M	!+Update+!	update	The setting of the update selector switch. Values correspond to switch nomenclature.
T	!+UpdATTW=Other+!	boolean	!+Update+! != $Flyover$ & !+Update+! != HUD & !+Update+! != $Radar$ & !+Update+! != $TacL-L$

TABLE 153. Variable Dictionary (Sheet 20 of 22)

MCT	Name	Type	Interpretation
M	!+USC az on ASL+!	angle	If !+ASL rotation+! = 90 or 270 then !+ALS rotation+!. Otherwise, the azimuth angle of a point on the HUD azimuth steering line (ASL) symbol at elevation equal to !+USC elevation+!. The ASL symbol is taken to be a line segment arbitrarily long; if no point on the actual symbol is at the given elevation, the result will be calculated as though the segment were long enough to reach that elevation. The resolution of !+USC elevation+! is assumed not to be less than #HUD symbol el res#. The computed result is limited to be within the HUD field of view; #HUD symbol az max# >= !+USC az on ASL+! >= #HUD symbol az min#).
C	!+USC azimuth+!	angle	Present angle between the ORA and the projection of the ray between the HUD origin and the symbol onto the plane formed by the ORA and the !!horizontal ORA!! or (if the symbol is currently off) the azimuth angle at which the symbol would be displayed. The angle is positive (negative) if to the right (left) of the positive ORA as viewed by the pilot.
C	!+USC elevation+!	angle	Present angle between the ORA and the projection of the ray between the HUD origin and the symbol onto the plane formed by the ORA and the !!vertical ORA!! or (if the symbol is currently off) the elevation angle at which the symbol would be displayed. The angle is positive (negative) if above (below) of the positive ORA as viewed by the pilot.
MC	!+USC mode+!	HUD_ind_cntrl	Present display mode of the symbol.
MC	!+V coarse bias+!	accel	Measurement error for the Zp axis when the velocities are being measured by the coarse scale.
MC	!+V coarse scale+!	speed	Scale factor per pulse used for velocity calculation for the Zp axis when the velocities are being measured by the coarse scale.
M	!+velocity east system+!	speed	The aircraft's current east velocity component, computed from the best available sources.
M	!+velocity north system+!	speed	The aircraft's current north velocity component, computed from the best available sources.
M	!+velocity vertical system+!	speed	The aircraft's current vertical velocity component, computed from the best available sources.
MC	!+VV mode+!	HUD_ind_cntrl	Present display mode of the symbol.
T	!+Walleye+!	boolean	!+Weapon Class+! = WL
T	!+WDMFS+!	boolean	!+natt+! OR !+boc+! OR !+ccip+!
M	!+Weap Interval+!	integer	The value set on the switch labeled "Interval-Ft"; integer between zero and #weap interval max#, inclusively.
M	!+Weap Pairs+!	boolean	$true$ iff the switch labeled "Pair"/"Single"/"Simult Rkts" is currently set to the "Pair" position.
M	!+Weap Quantity+!	integer	The value set on the switch labeled "Quantity"; integer between 0 and #weap quantity max#, inclusively.
M	!+Weapon Class+!	weap_class	The class of the weapon loaded on the currently active weapon station(s). If no weapon station is active, then GN.

TABLE 153. Variable Dictionary (Sheet 21 of 22)

MCT	Name	Type	Interpretation
M	!+Weapon Mode+!	weap_mode	The current setting of the master function switch (weapon mode).
M	!+WEAPTYP+!	integer	The current value of the physical input /WEAPTYP/, represented as an integer.
M	!+wind vel+!	velocity	The current velocity of the wind, measured from the best available sources. The length of the vector (always non-negative) is equal to the speed of the wind. The direction of the vector is the direction that the wind is blowing from in the East-North-Vertical reference frame.
M	!+WOG+!	boolean	$true$ iff weight on landing gear detected.
M	!+wpns rlsd+!	integer	The number of weapons in the current stik that have actually been released. Undefined if no stik currently exists.
C	!+X coarse rotation+!	angle	The inertial platform is rotated about the specified axis by angle !+X coarse rotation+!. A positive (negative) angle causes the platform to rotate CW (CCW) looking along the axis from the origin.
MC	!+X corr increm+!	angle	Size of rotation increments per pulse for the Xp axis during fine correction.
MC	!+X drift+!	angrate	Drift rate for the Xp axis.
C	!+X fine rotation+!	angle	The inertial platform is rotated about the specified axis by angle !+X fine rotation+!. A positive (negative) angle causes the platform to rotate CW (CCW) looking along the axis from the origin.
M	!+XACC test result+!	boolean	true iff the accelerometer and torque registers associated with X axis of the IMS pass built-in test.
C	!+Y coarse rotation+!	angle	The inertial platform is rotated about the specified axis by angle !+Y coarse rotation+!. A positive (negative) angle causes the platform to rotate CW (CCW) looking along the axis from the origin.
MC	!+Y corr increm+!	angle	Size of rotation increments per pulse for the Yp axis during fine correction
MC	!+Y drift+!	angrate	Drift rate for the Yp axis.
C	!+Y fine rotation+!	angle	The inertial platform is rotated about the specified axis by angle !+Y fine rotation+!. A positive (negative) angle causes the platform to rotate CW (CCW) looking along the axis from the origin.
M	!+YACC test result+!	boolean	true iff the accelerometer and torque registers associated with Y axis of the IMS pass built-in test.
C	!+Z coarse rotation+!	angle	The inertial platform is rotated about the specified axis by angle !+Z coarse rotation+!. A positive (negative) angle causes the platform to rotate CW (CCW) looking along the axis from the origin.
MC	!+Z corr increm+!	angle	Size of rotation increments per pulse for the Zp axis during fine correction
MC	!+Z drift+!	angrate	Drift rate for the Zp axis.
C	!+Z fine rotation+!	angle	The inertial platform is rotated about the specified axis by angle !+Z fine rotation+!. A positive (negative) angle causes the platform to rotate CW (CCW) looking along the axis from the origin.
M	!+ZACC test result+!	boolean	true iff the accelerometer and torque registers associated with Z axis of the IMS pass built-in test.
T	!A/C facing target!	boolean	TRUE if !+gr ac tgt+! < 90 OR !+gr ac tgt+! >270

TABLE 153. Variable Dictionary (Sheet 22 of 22)

MCT	Name	Type	Interpretation
T	!fly-to point!	fly-to-state	$Dest$ or $Mark$ indicated by the settings of !+Fly to state+! and !+Fly to num+!.
M	!ground range!	distance	Horizontal distance to some point.
M	!low drag!	boolean	!+low drag release+!
T	!OAP!		Offset aimpoint - used in *BOCoffset*, *Noffset*, *HUDdown2*, *SBOCoffset*, *Snoffset*, and *SHUDdown2* modes. The OAP is a landmark close to a target, and the target is defined by its position relative to the OAP. In these modes, the !Fly-to point! is the original OAP, which may be changed by slewing.
M	!OTS!	boolean	!+OTS+!
M	!Ready Station REQ+!	boolean	!+Station selected=! & !+Weapon Class+! != UN & !+Weapon Class+! != GN
M	!Station selected+!	boolean	!+nbr rdy sta+! > 0
M	!system pitch!	angle	!+pitch system+!
T	!target!		See Table 154 on page 150.

TABLE 154. Target in Weapon Delivery Modes

MODES	!target!
Nattack *BOCFlyto0* *Snattack* *SBOCFlyto0*	location on ground overlaid by AS at designation and after each slew
HUDdown1 *SHUDdown1*	location on ground intersecting the HUD ORA at designation
HUDdown2 *SHUDdown2*	OAP is the point on the ground intersecting the HUD ORA at designation; target is a point on the line drawn from the OAP in the direction defined by !Offset bearing!; the target is !Offset range! nmi from the OAP.
Noffset *Snoffset*	OAP is the point on the ground overlaid by AS at designation and after each slew; target is a point on the line drawn from the OAP in the direction defined by !Offset bearing!; the target is !Offset range! nmi from the OAP.
BOC *SBOC*	Before slewing, the target is the !Fly-to point!; after slewing, it is the point defined by the radar cursors when @T(!slewing finished!) occurs.
BOCoffset *SBOCoffset*	OAP is defined in the same way the target is defined *BOC* (*SBOC*); target position relative to OAP defined in *Noffset* (*Snoffset*)
Walleye	location on ground overlaid by AS at designation